

Change Table:

Date of Change	Related Issue Number	Updated by	Revision
11/16/99	98	C.Collins	Added Section 5.5.3 - All documents to be posted to the WDTIP web site will come directly to the Project Controller. The PC will determine that the document version is the most current and that the document is located within the Approved folder on the LAN. The PC will communicate directly with the WDTIP webmaster and inform him of the location (path) of the document. At that point, the WDTIP webmaster will initiate the State's approval process. Upon the State's approval, the webmaster will post the document to the web site.
2/25/00	128	F. Vranesic/ F. Ernst	Changed page 53-54 to reflect the manual deletion process used during Module Check-In. Updated page 56 to reflect the steps used in the promotion process using the Access tool.
9/20/00		R. Frey	Global changes made to rename CITS (Configuration Item Tracking System) to PTS (Project Tracking System). Other changes made based on comments identified in the Configuration Management Plan ver 6 comments matrix.

WELFARE DATA TRACKING IMPLEMENTATION PROJECT

CONFIGURATION MANAGEMENT PLAN

Table Of Contents

1	INTRODUCTION	6
1.1	PURPOSE.....	6
1.2	OBJECTIVE.....	6
1.3	SCOPE.....	6
2	REFERENCES	7
2.1	REFERENCE LIST.....	7
2.1.1	Project References	7
2.1.2	Non-Project References	7
2.2	DEFINITIONS AND ACRONYMS.....	8
3	CONFIGURATION MANAGEMENT ORGANIZATION.....	11
3.1	CONFIGURATION MANAGEMENT RELATIONSHIP	11
3.2	PURPOSE AND ACTIVITIES.....	11
3.2.1	Configuration Control Board.....	12
3.2.2	Project Management.....	12
3.2.3	Configuration Management Team.....	12
3.2.4	Quality Assurance Team.....	12
3.2.5	Team Leads	12
3.2.6	Project Management Support Team.....	13
3.2.7	Application Team.....	13
3.2.8	Implementation Team.....	13
3.2.9	Independent Verification and Validation Vendor Team (IV&V)	13
3.2.10	Configuration Functions.....	13
4	CONFIGURATION IDENTIFICATION.....	15
4.1	ITEM CLASSIFICATION	15
4.2	VERSIONING.....	17
4.3	BASELINE.....	18
4.3.1	Baseline Classification.....	19
4.4	ARCHIVING.....	20
4.5	BACKUP AND RECOVERY	20
4.6	PHYSICAL COMPUTING ENVIRONMENT.....	21
4.6.1	Software Storage.....	22
4.6.2	Document Storage.....	23
4.7	CONFIGURATION NAMING CONVENTION.....	23
4.7.1	Software Naming Convention.....	23
4.7.2	Document Naming Convention.....	24
4.8	PROGRAMMING AND TESTING STANDARDS.....	24
5	CONFIGURATION CONTROL.....	24
5.1	BASELINE PROCESS	24
5.1.1	Baseline Security.....	24
5.1.2	Starting the process	25
5.1.3	Baseline Creation.....	25
5.1.4	Baseline Completion.....	25
5.2	ARCHIVING PROCESSES.....	25
5.2.1	Archive Security.....	26
5.2.2	Starting the process	26
5.2.3	Archive Creation	26

5.2.4	Archive Completion.....	26
5.3	CONFIGURATION CONTROL PROCESS.....	26
5.3.1	Configuration Control Process Flow.....	27
5.3.2	Change Request Submission.....	29
5.3.3	Configuration Control Board Processes.....	29
5.3.3.1	Definition of Statuses for Change Request.....	29
5.3.3.2	Definition of Priority Levels for Change Requests.....	30
5.3.4	Quick Fix Process.....	31
5.4	SOFTWARE CONTROL PROCESS.....	31
5.4.1	Promotion Model.....	31
5.4.2	Starting the Process.....	32
5.4.3	Software Migration Process.....	35
5.4.4	Migration Completion.....	38
5.4.5	Processing Rejected Items.....	38
5.5	DOCUMENT CONTROL PROCESS.....	38
5.5.1	Document Control Process Model.....	38
5.5.2	Document Control Stages.....	38
5.5.3	Web Site Controls for Documentation.....	39
5.6	DATA CONTROL PROCESS.....	39
5.7	INCIDENT TRACKING PROCESS.....	40
5.8	CONFIGURATION SECURITY.....	41
6	CONFIGURATION STATUS ACCOUNTING.....	43
6.1	LOGS.....	43
6.2	REPORTS.....	43
6.3	ACCOUNTING PROCESS.....	43
7	CONFIGURATION AUDITING.....	44
8	CONFIGURATION TOOLS.....	45
8.1	LAN (PC-BASED) TOOLS.....	45
8.1.1	Project Tracking System (PTS).....	45
8.2	MAINFRAME TOOLS.....	46
8.2.1	Library Management Tool.....	46
8.2.1.1	Terminology.....	46
8.2.1.2	Source Library Management.....	47
	Copybooks / CICS maps.....	47
	Copybooks / CICS maps.....	47
	Module Development Tracking.....	48
	Module Check-Out.....	49
	Module Check-Out Security.....	50
	Module Check-In.....	50
	Module Check-In Security.....	52
	Module Promotion.....	53
	Module Promotion Tracking.....	55
	Module Promotion Security.....	55
8.2.1.3	Scheduled Backup.....	55
8.2.2	Platinum Database Utilities.....	55
8.2.3	RACF.....	56
9	APPENDICES.....	57
9.1	FORMS.....	57
9.1.1	Configuration Item Change Request Form.....	58
9.1.2	Data Model Change Request Form.....	59
9.1.3	Database Change Request Form.....	60

9.1.4	Quality Assurance Review.....	62
9.1.5	Move Request Form.....	63
9.2	DELIVERABLES ACCEPTANCE CRITERIA.....	64

List of Figures

<i>Figure 1: Configuration Management Relationship.....</i>	<i>11</i>
<i>Table 1: Configuration Item Classification.....</i>	<i>15</i>
<i>Figure 2: Baselines in Relation to Project Life Cycle.....</i>	<i>19</i>
<i>Figure 3: WDTIP Physical Computing Environment</i>	<i>22</i>
<i>Table 2: Sample WDTIP File Directory Structure</i>	<i>23</i>
<i>Figure 4: Configuration Control Process Flow</i>	<i>28</i>
<i>Table 3: Definition of Priority Levels for Change Requests</i>	<i>30</i>
<i>Figure 5: WDTIP Promotion Model</i>	<i>32</i>
<i>Figure 6: WDTIP Document Control Process Model.....</i>	<i>38</i>
<i>Figure 7: Reference Module Copying Process</i>	<i>49</i>
<i>Figure 8: Development Environment Reference Library Check-in Process</i>	<i>51</i>
<i>Figure 9: Reference Module Promotion Process</i>	<i>54</i>

List of Tables

<i>Table 1: Configuration Item Classification.....</i>	<i>15</i>
<i>Table 2: Sample WDTIP File Directory Structure</i>	<i>23</i>
<i>Table 3: Definition of Priority Levels for Change Requests</i>	<i>30</i>

1 Introduction

1.1 Purpose

Configuration Management (CM) is a formal control discipline which provides the Welfare Data Tracking Implementation Project (WDTIP) Team with the methods and tools to identify documentation and software developed, establish baselines, control changes to these baselines, record and track status, and audit the deliverables and products. The Configuration Management Plan (CMP) identifies the CM processes as well as outlines the change control procedures of the project that will be necessary during the development of the TRAC Application.

The CMP is a living document and will be updated as necessary. The document is intended to be used as reference for all project participants.

1.2 Objective

The objective of the CMP is to provide a framework to manage and control the product of the project. The CMP describes these processes and methods that identify and control the WDTIP baselines. These include corresponding software (i.e., application, third party, test, conversion, and other software), project documentation, and discrepancies, as well as changes that impact the project.

The goal of the WDTIP CMP is to keep the control process simple, yet establish mechanisms to address complex project modifications should they arise, and normal changes to deliverables. The intent is to allow for successful implementation of the WDTIP using the CMP to manage the flow of desired changes and only implement approved changes.

1.3 Scope

The CMP addresses the process used to control the System Development Life Cycle (SDLC) of the WDTIP. The CMP will provide strong control during all phases of the project. The scope of CM includes configuration reference documents, management organization, identification, control, status accounting, auditing, and tools. In addition, the scope of CM will be limited to the configuration items (CI) identified in the **Configuration Identification** section of this document.

2 References

The Reference List section identifies all internal and external documents that are referenced in the CMP. The Definitions and Acronyms section provides a list of configuration management-related definitions and acronyms used throughout the CMP document.

2.1 Reference List

Project and non-project references are listed below:

2.1.1 Project References

1. WDTIP Consultants Guidelines, “Acronyms List” section (most current version)
2. **Project Management Plan** (most current version)
3. **Design/Coding Standards Guide** (most current version)
4. **System Architecture Model** (most current version)
5. **Completed Source Modules/Unit Test** (most current version)
6. **IV&V Plan** (most current version)
7. **Transition Plan** (most current version)

2.1.2 Non-Project References

1. Deloitte Consulting’s *Framework for Computing Solutions* published by Deloitte Consulting
The Software Configuration Management Plan - model text
Web site: <http://www.airtime.co.uk/users/wysywig/cmp.htm>
2. IEEE std. 828- 1990, Standard for Software Configuration Management Plans
3. IEEE std. 1042- 1987, Guide to Software Configuration Management
4. IEEE std. 610.12- 1990, IEEE Standard Glossary of Software Engineering Terminology
5. MIL-STD-498, Software Development and Documentation, December 5, 1994

2.2 Definitions and Acronyms

The following is a selected list of definitions and acronyms used within the CMP. The purpose of this list is to provide a clear definition of the terms used throughout the body of the document. Standard definitions were drawn from the IEEE Standard Glossary of Software Engineering Terminology (IEEE Std 610.12 – 1990) and from Military Standards (MIL-STD-498). Project specific definitions and acronyms (WDTIP Consultants Guidelines) have been added to this list.

Allocated Baseline – The initial approved specifications governing the development of configuration items that are part of a higher level configuration item. *Contrast with: functional baseline; product baseline.*

Archive – Archiving is a process of storing Configuration Items in a secure manner. The purpose of archiving is to provide recoverability to a past state. Although the process for creating an archive is similar to that of taking a baseline, the method of storage for both is different. Whereas baselines are maintained in easily accessible media for reference during the project lifecycle, archives are stored on secure media, often with the additional precaution of offsite storage to ensure they are not lost.

Backup – A process of creating a copy of a configuration item to prevent the loss of work.

Baseline – (1) A specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for further development, and that can be changed only through formal change control process. (2) A document or set of such documents formally designated and fixed at a specific time during the life cycle of a configuration item.

Configuration Control – An element of configuration management, consisting of the evaluation, coordination, approval or disapproval, and implementation of changes to configuration items after formal establishment of their configuration identification. *Syn: change control.*

Configuration Control Board (CCB) – A group of people responsible for evaluating and approving or disapproving proposed changes to configuration items, and for ensuring implementation of approved changes. *Syn: change control board.*

Configuration Identification – An element of configuration management, consisting of selecting the configuration items for a system and recording their functional and physical characteristics in technical documentation.

Configuration Item (CI) – An aggregation of hardware, software, or both, that is designated for configuration management and treated as a single entity in the configuration management process.

Configuration Item Change Request - A formal request that defines a change to any component under Configuration control.

Configuration Management (CM) – A discipline applying technical and administrative direction and surveillance to: identify and document the functional and physical characteristics of a

configuration item, control changes to those characteristics, record and report change processing and implementation status, and verify compliance with specified requirements.

Configuration Management Plan (CMP) - The document that defines the process to be followed in identifying, tracking and controlling changes to configuration items.

Configuration Status Accounting – An element of configuration management, consisting of the recording and reporting of the information that is needed to manage a configuration effectively. This information includes a listing of the approved configuration identification, the status of proposed changes to the configuration and the implementation status of the approved changes.

Database – A collection of interrelated data stored together in one or more computerized files.

Deliverable - The tangible result of completion of a task or set of tasks. A deliverable may be as simple as a completed form or letter or as complex as a full solution.

Environmental Software - Includes operating system software, language compilers and other software used to support the general computing environment.

Function – A defined objective or characteristic action of a system or component. For example, a system may have inventory control as its primary function.

Functional Baseline – The initial approved technical documentation for a configuration item.
Contrast with: allocated baseline; product baseline.

Incident – A program logic anomaly that is identified in the TRAC Application during integration/system and user acceptance testing phases of the project.

May - Indicates an item or activity appropriate under some, but not all, conditions; for which there are a number of acceptable alternatives; or for which there is no professional consensus.

Milestone - Denotes a particular event in time when a specific deliverable product is to be delivered or an important event is to occur.

Product Baseline – The initial approved technical documentation (including, for software, the source code listing) defining a configuration item during the production, operation, maintenance, and logistic support of its life cycle. *Contrast with: allocated baseline; functional baseline.*

Reject - For the purposes of this document, a disapproval is synonymous with a reject.

Release - A collection of related configuration items that are each uniquely identified by a version number.

Shall - Indicates an item or activity is required.

Should - Indicates a recommended item or activity.

Version – (1) An initial release or re-release of a computer software configuration item, associated with a complete compilation or recompilation of the computer software configuration item. (2) An initial release or complete re-release of a document, as opposed to a revision resulting from issuing change pages to a previous release.

Version Control - A systematic process that captures and maintains accurate date and content information of the successive versions of the configuration items.

Welfare Data Tracking Implementation Project (WDTIP) – The official name of the project that is the subject of this document.

Will - Indicates an item or activity is a goal, which may or may not be attainable. See Shall.

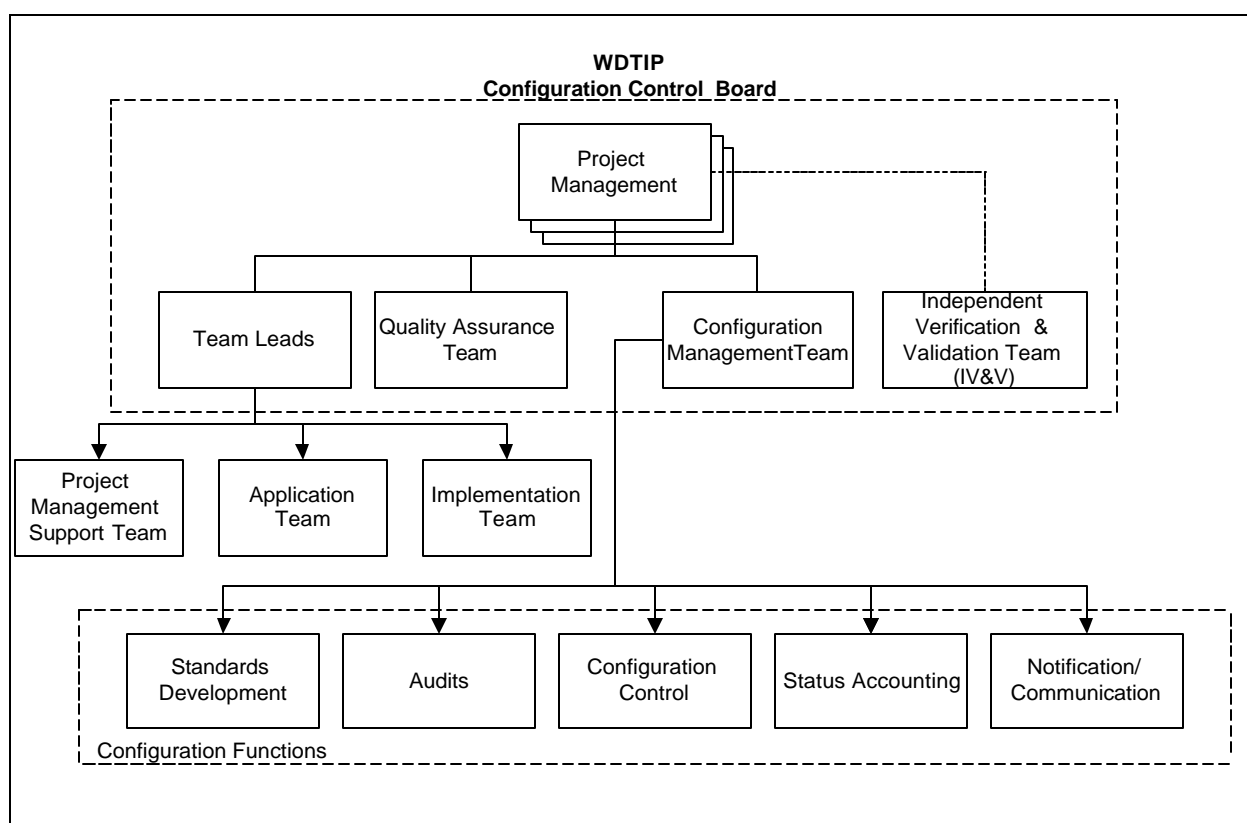
3 Configuration Management Organization

An important element of the CMP is full integration of CM into the WDTIP organizational structure. This structure ensures clear lines of authority and responsibility for the administrative, technical, and supporting tasks defined and scheduled for the project.

3.1 Configuration Management Relationship

The configuration management relationship to the project management structure is illustrated in **Figure 1** below:

Figure 1: Configuration Management Relationship



3.2 Purpose and Activities

CM is a discipline to 1) ensure the proper identification of the configuration, 2) control changes, and 3) record and communicate the change implementation status of the physical and functional characteristics of the WDTIP. CM tracks what is required, designed, and produced by the

project; evaluates changes including effects on technical and operational performance; and conducts reviews and walkthroughs throughout the system development life cycle.

The following are the components for the configuration management relationship of the WDTIP (refer to the **Organizational Charts** section within the **Project Management Plan** deliverable):

3.2.1 Configuration Control Board

A reviewing and controlling body that exercises established processes to classify, approve or reject, release, implement and confirm changes to agreed specifications and baselines. The Configuration Control Board (CCB) will be involved in all review processes and change management for the project through structured and strictly defined processes that are identified in the Plan and will ensure that all documents/deliverables are reviewed and signed off before they are version controlled.

The WDTIP project management (i.e., co-project managers from both the State and Deloitte Consulting) will serve as the chair in the highest level on the Board, and will have the authority to make decisions. The Team Leads, Quality Assurance Team, IV&V, and Configuration Management Team operate at a level below the chair and serve to make decisions by functioning as a body.

3.2.2 Project Management

The role focuses on the long-term vitality of the organization, rather than the day-to-day project concerns that fall on the Team Leads. Project Management has oversight over all team members and is responsible for enforcing project policies.

3.2.3 Configuration Management Team

The Configuration Management Team (CMT) is co-managed by the Technical Support Team Lead and the Project Controller. The Technical Support Team Lead has responsibility for all Configuration Items with the exception of Category II items. The Project Controller has responsibility for Category II items. Refer to the **Configuration Identification** section of this document for category definitions. The CMT will administer the technology and deliverable documentation standards of the project to all team members.

3.2.4 Quality Assurance Team

The Quality Assurance (QA) Team has two separate entities – internal and external. Internal QA engages in ongoing internal examination of a work product/process or set of work products/processes to assess compliance with specifications, standards, contractual agreements, or other criteria within the WDTIP. External QA has a role similar in nature to internal QA. External QA occurs at a minimum of once for each phase of the project. Refer to the **Quality Assurance Approach and Plan** section of the **Project Management Plan** deliverable.

3.2.5 Team Leads

The team lead is a person that oversees a number of persons associated together in work or activity within the WDTIP. Refer to the **Organizational Charts** section within the **Project Management Plan** deliverable.

3.2.6 Project Management Support Team

A number of persons that support Project Management and Team Leads in carrying out project objectives.

3.2.7 Application Team

A number of persons that are involved in requirements' analysis, design, development, testing, data modeling, database administration, and internal quality assurance of the TRAC Application.

3.2.8 Implementation Team

A number of persons that are involved in developing a stakeholder communication plan, training, implementation support, and user acceptance testing for the project.

3.2.9 Independent Verification and Validation Vendor Team (IV&V)

Deliverable Validation and Verification is a disciplined approach to assessing products throughout the project. Validation and Verification provides management with insights into the state of the project and products, allowing for timely change in the products or in the development and support processes. Refer to the **IV&V Plan** deliverable.

3.2.10 Configuration Functions

Below are the configuration functions that are overseen by the CMT. Included in this subsection is a brief description of each configuration function.

- ❑ **Standards Development**

The process of establishing a model or rule for the WDTIP to follow throughout the life of the project. It is also the discipline for identifying the configuration of an item, and documenting its functional and physical characteristics.

- ❑ **Audits**

The process of reviewing the project team's work in relation to technical/functional requirements, compliance with project standards, and to suggest alternatives and/or approaches.

- ❑ **Configuration Control**

Refer to the **Definitions and Acronyms** section of this document for a description of Configuration Control.

- ❑ **Status Accounting**

The process of recording, documenting, and reporting the current status of all Configuration Items (CI). Refer to the **Configuration Status Accounting** section of this document for further information on status accounting.

□ **Notification/Communication**

Written or printed matter that gives notice to related members of the WDTIP. Also, a process by which information is exchanged between team members.

4 Configuration Identification

Configuration Identification consists of defining and identifying items subject to configuration management. This section will first identify and classify all WDTIP Configuration Items (CI) into separate categories. In addition, this section will also describe the versioning, baselining, archiving, backup and recovery, physical storage environment, naming convention, programming, documentation, and testing standards of these CI. This section of the document describes the characteristics of several configuration activities. Detailed processes outlining the performance of those activities are included in the **Configuration Control** section of this document.

4.1 Item Classification

Configuration Items (CI) are grouped together into various categories for identification and control purposes. Processes defined in this document can be applied to the members of a category uniformly. The criteria for inclusion of an item into a category are described in the following table.

Table 1: Configuration Item Classification

Category	Title	Definition
I	Application Software	Software developed as part of the WDTIP contract. The items in this category are operating components of the TRAC Application. This will include, but not be limited to, the following groups of items: <ul style="list-style-type: none">• Source Code• Database Objects (DB2 tables, indexes, etc...)• CICS Maps• Copybooks• DCLGENs• JCL• JCL Control Cards• Database Bind Cards• Database Reference Modules (DBRM)• Batch Program Load Modules• CICS Program Load Modules

Category	Title	Definition
II	Project Documentation	<p>Documents developed as part of the WDTIP contract. The items in this category define and describe the project. This will include, but not be limited to, the following groups of items:</p> <ul style="list-style-type: none"> • Monthly Status Reports • Project Management Plan • Updated Stakeholder Communication Plan • Configuration Management Plan • Updated Business Requirements Document • Design/Coding Standards Guide • System Architecture Model • Implementation Strategy • Detailed Design Specification Document • WDTIP Phase II Workplan • Completed Source Code/Unit Test Plans • Implementation Plan • Integration/System Test Sign-Off • Training Curriculum • WDTIP Phase III Workplan • User Acceptance Test Plan and Scenarios • System Documentation • Transition Plan (M&O) <p>All sub-components of the deliverables listed above can be found in the Deliverable Acceptance Criteria subsection of the Appendix section in this document.</p> <p>In addition to the documents listed above, other non-contractual, but deliverable-like documents such as newsletters should be considered part of Category II items for configuration control.</p>
III	Production Data Files	<p>Production data that is actively managed by the project. This includes source data from the county systems as well as processed data residing in the central database of the project. Production data used in testing should also be considered part of Category III items for configuration control.</p>

Category	Title	Definition
IV	Third Party Software	<p>Software that already exists as a vendor provided product and is maintained by a vendor. This category of software may be deliverable or non-deliverable software. Use of items in this category will be tracked as they pertain to the items in categories I, II, and III. An example of a third-party software is Erwin.</p> <p>Analysis and reporting shall be made through the CCB in instances of functional changes in third-party software. An example of this relationship would be a Database Object (Category I) and its related database product (DB2).</p>
V	Internet Web Pages	<p>Published information referring to the project. Documents in this category may be based upon controlled documents. Content of the Internet pages themselves is under State control.</p>

CM processes are defined for Category I, II, and III items. CM will use processes assigned by the vendor for Category IV items and HHSDC LAN Support for Category V items.

4.2 Versioning

When a change is required of a managed item, a new version shall be created. Version control of CI is applied in both the LAN and mainframe environments. The responsibility and authority for the creation of versions is described in the **Configuration Security** section of this document. Different processes will be applied to the various categories of CI. The following describes how versioning will be applied to different configuration item categories:

□ **Category I**

Category I source code modules on the mainframe will be controlled by a library management tool that assigns version numbers automatically. The library management tool will control source code, CICS maps, copybooks, DCLGENs, JCL, JCL and control cards. Items not controlled by the tool (database objects, DBRMs, batch & CICS load modules) will be versioned by the Technical Support Team as part of the software acceptance and migration process (refer to the **Software Control Process** section of this document).

Versioning will occur when an item is migrated at the direction of the CCB. As an item is promoted into new environments, existing items in the receiving environment are copied and saved so that recovery to a previous state is always possible.

❑ **Category II**

The Project Controller will control all Category II items on the LAN during the submission, acceptance and subsequent modifications. As a document or file is migrated into a new directory, a version number will be appended to its long file name. This process is performed manually by the Project Controller as part of the migration procedure. For further details, refer to the **Document Control Process** section of this document.

❑ **Category III**

Data in all files will be stored in file structures that allow recovery to previous states for a specified time period. The full range of items included in this category will be identified during the detailed system design. Each item included in this category will have a unique requirement for recoverability. The requirement for recoverability and the versioning characteristics of the item will be defined at the point of identification during the detailed system design.

❑ **Category IV and V**

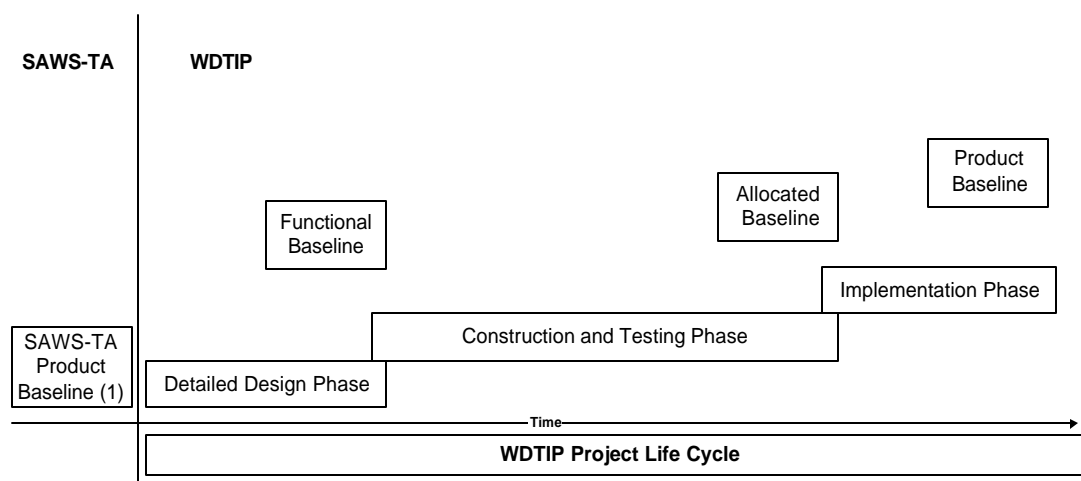
Category IV and V items do not fall under the version control of the project. Changes in the versions of these items will be tracked as they pertain to controlled items when they occur. Recovery to previous states will only be possible to the extent provided by the vendor or other controlling organization for these items.

4.3 Baseline

Baselining is a process designed to provide a fixed reference source for a point of consistency within the project lifecycle. Baselines tie documentation, programs and supporting files together. In the context of this project, the items being stored are a related sequence of software and documentation drawn from Configuration Categories I and II. The data contents of files and databases that are included in Category III are not included in the baselines. The process of saving that information is described in the **Backup and Recovery** section of this document.

Baselines define a state of the system at a specific point in time. In WDTIP, three baselines are planned: Functional, Allocated, and Product. Each baseline coincides with the conclusion of a major phase in project life cycle. This relationship is shown in the following figure.

Figure 2: Baselines in Relation to Project Life Cycle



1. *WDTIP uses selected items of the SAWS-TA project as a baseline to build on. Elements of the SAWS-TA Product Baseline contribute to an understanding of the requirements for the TRAC Application. The SAWS-TA project was completed prior to the start of WDTIP. The product baseline of the previous project is shown as a reference point in the baseline diagram.*

4.3.1 Baseline Classification

The Functional Baseline will be established at the conclusion of the Detailed Design Phase of the project. It will include, but not be limited to the following documentation:

- ❑ **Updated Business Requirements Document**
- ❑ **Project Management Plan**
- ❑ **Updated Stakeholder Communication Plan**
- ❑ **System Architecture Model**
- ❑ **Detailed Design Specifications Document**
- ❑ **Design/Coding Standards Guide**
- ❑ **Implementation Strategy**
- ❑ **Configuration Management Plan**
- ❑ **Supporting documents**

ex: formal review comments, e-mail, acceptance letters, etc...

These documents describe capabilities representing specific functions the system shall perform or provide in accomplishing the tasks and activities required of the TRAC Application.

The Allocated Baseline will be established at the conclusion of the Construction and Testing Phase of the project. It will contain documentation and software that fulfill the requirements established at the Functional Baseline. Additional functionality captured at this baseline will include, but not be limited to, the following areas:

- ❑ **Completed Source Code/Unit Test Plan**
- ❑ **Implementation Plan**

- ❑ Training Curriculum
- ❑ Integration/System Test Plan and Sign-Off
- ❑ Supporting documents
ex: formal review comments, e-mail, acceptance letters, etc...

The Product Baseline is established at the end of the Implementation Phase. It will include the final output of the project. Information contained in this baseline will include, but not be limited to, the following areas:

- ❑ User Acceptance Test Sign-Off
- ❑ System in Production
- ❑ Conversion Loads
- ❑ User Training
- ❑ Transition Plan
- ❑ Supporting documents
ex: formal review comments, e-mail, acceptance letters, etc...

4.4 Archiving

Archiving is a process of storing CI in a secure manner. In the context of this project, the items being stored are a related sequence of software and documentation drawn from Configuration Categories I and II. The data contents of files and databases that are included in Category III are not included in the archives. The process of saving that information is described in the **Backup and Recovery** section of this document.

The process for creating an archive is similar to that of taking a baseline, but the purpose differs. Where the baseline is designed as a reference source for a point of consistency within the project lifecycle, the archive is designed to provide recoverability to a past state.

A major difference in these two focuses is the method of storage. Baselines are maintained in easily accessible media for reference during the project life cycle. Archives are stored on secure media, often with the additional precaution of offsite storage to ensure they are not lost.

4.5 Backup and Recovery

Backup and recovery of the WDTIP CI will be maintained by HHSDC. Technical Support will review existing HHSDC process for backup and recovery to determine whether or not it adequately meets project needs. If backup and recovery at HHSDC is determined to be insufficient for the project needs, project management (includes both State and Deloitte Consulting) will work together to come up with a solution. In addition, project management will include all appropriate internal and external stakeholders in the process of coming up with a solution.

❑ **Category I Items**

Category I items are all located on storage devices maintained by the HHSDC. The backup and recovery procedures for these devices are part of operational practices and procedures of the data center.

❑ **Category II Items**

Category II items are located on the public (H:) drive on the HHSDC LAN. The HHSDC LAN is backed-up onto tape on a nightly basis by the LAN administrator in accordance with established HHSDC practices. The tape is locked up with the SAWS Department and any recovery of lost material can be made via the LAN administrator.

❑ **Category III Items**

Technical Support manages the data content of the project databases in all environments. The full content of active DB2 databases is backed up on a nightly basis by scheduled utility applications. Inactive or unused databases are backed up on a weekly basis by the same process. The backup utility used provides a consistent point of recovery for the information contained in the databases. All database backups are executed by an automated scheduling system and operate normally without the direct intervention of project personnel.

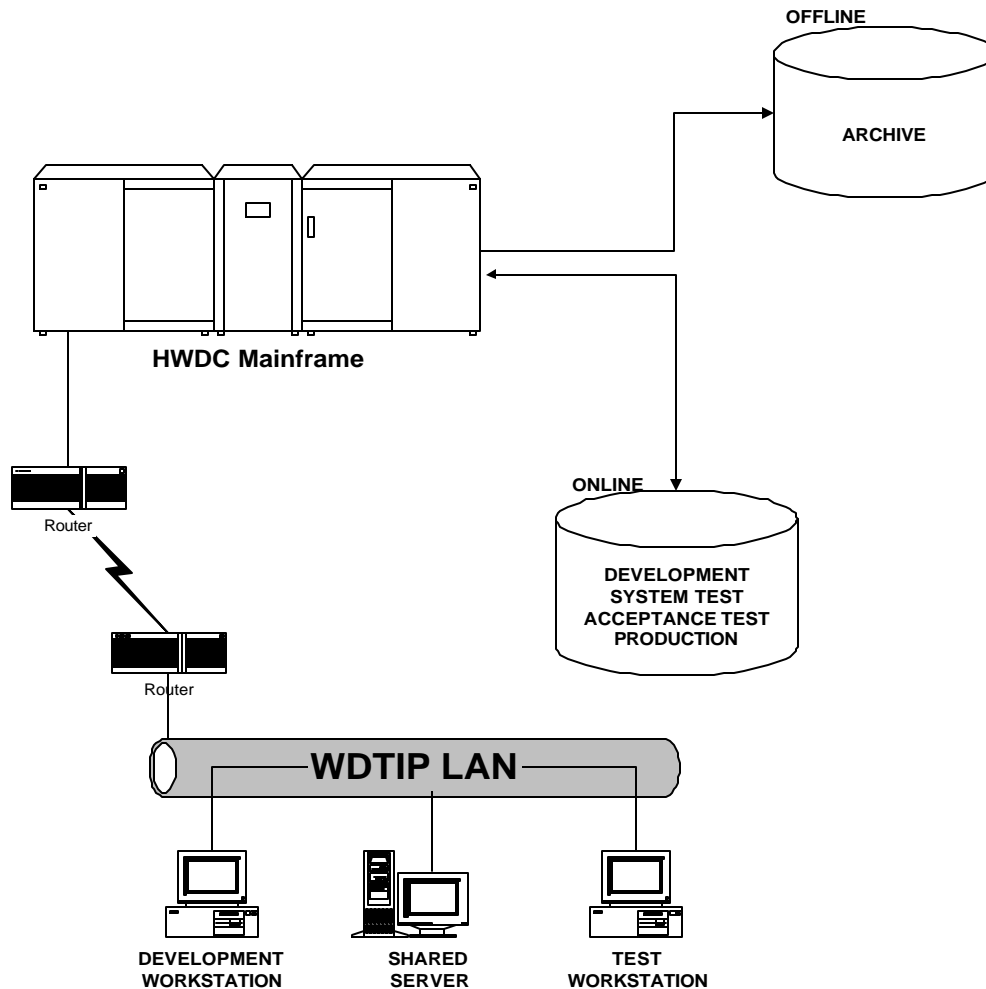
4.6 Physical Computing Environment

This section provides a high level description of the physical computing environment utilized by the project. For further details, refer to the **System Architecture Model** deliverable.

The WDTIP physical computing environment resides on the computing systems maintained at HHSDC. The data center operates interconnected mainframe computers with associated direct and sequential access devices. The capacity of the data center is designed to meet the combined requirements of all client organizations. The project offices are connected to HHSDC via a LAN system.

The following figure depicts the major components of the physical computing environment.

Figure 3: WDTIP Physical Computing Environment



4.6.1 Software Storage

All Category I software development and testing activities are performed in the mainframe environment. Shared access to the environment is provided through LAN connected workstations in the project offices. Libraries are defined to contain software items according to the Naming Standards section of the **Design and Coding Standards Guide**. These libraries are protected from failure by the storage protection processes of HHSDC. Please reference HHSDC storage backup procedures for further information.

DB2 databases supporting the TRAC Application will reside exclusively on HHSDC media. They will share the same level of protection that the application software libraries enjoy. Additional protection against loss for the data content of the database is provided by scheduled backup processes. Further details about those processes are provided in the **Backup and Recovery** section of this document.

4.6.2 Document Storage

All Category II items will be stored on a shared network drive that can be accessed by all team members. A project file directory structure on the LAN has already been established to facilitate the logical storage and retrieval of project Category II items. Category II items are organized into different functional areas (e.g., Deliverables, Forms, Misc., etc...) as separate folders. These functional areas can be further organized into sub-folders for better organization. Any team can create new folders and subfolders.

The current WDTIP file directory structure on the LAN is described below. Main WDTIP folders and sub-folders can be created, deleted, or modified throughout the life of the project. However, only the Project Controller can have write access to folders and subfolders containing officially submitted and approved deliverable documents. This will ensure the proper security of official project documents. The Project Controller will be responsible for maintaining the project

Table 2: Sample WDTIP File Directory Structure

Functional Area	Main WDTIP Folders	Purpose
Application Team	H:\WDTIP\Application Team	This folder contains items generated by the Application Team.
Deliverables	H:\WDTIP\Deliverables	This folder contains all project deliverables for all phases of the WDTIP.
Forms	H:\WDTIP\Forms	This folder contains all project forms.
Implementation Team	H:\WDTIP\Implementation Team	This folder contains items generated by the Implementation Team.
Miscellaneous	H:\WDTIP\Misc	This folder contains miscellaneous items.
Project Management	H:\WDTIP\Project Management	This folder contains items related to project management.
Technical Team	H:\WDTIP\Technical Team	This folder contains items generated by the Technical Team.
Workbench	H:\WDTIP\Workbench	This folder contains the Workbench tool and all information related to Workbench.

4.7 Configuration Naming Convention

This section will describe the standard naming conventions for documents (Category I), software (Category II), and baselines.

4.7.1 Software Naming Convention

Naming standards for Category I can be found in the following sections of the **Design and Coding Standards Guide**:

- ❑ Library and Naming Standards
- ❑ Program and Transaction Naming Standards

4.7.2 Document Naming Convention

All project staff shall use long file names to identify Category II items. For example, *Configuration Management Plan.doc* is used to identify that the file is the CMP deliverable.

4.8 Programming and Testing Standards

Programming and testing standards can be found in the following sections of the **Design and Coding Standards Guide**:

- ❑ COBOL Coding Standards
- ❑ JCL Standards
- ❑ Database Standards for DB2
- ❑ SQL Standards
- ❑ Screen Standards
- ❑ Report Standards
- ❑ Unit Testing Standards

5 Configuration Control

Configuration control is a detailed explanation of the processes developed to support the activities of baselining; archiving; requesting changes; software, document and data control; and incident tracking throughout the project life cycle. In addition, configuration control outlines the method to avoid the possibility of a change being implemented without due consideration of its effect on the baselines, including logistics impact, costs, schedules, performance, or interface with any external entities, etc.

5.1 Baseline Process

Baselines are a snapshot of a project at a specific and significant moment in time. They contain interrelated materials produced by the project such as documentation and software. The actual content of the baselines planned for the WDTIP is described in the **Configuration Identification** section of this document.

Three baselines are planned for this project: Functional, Allocated, and Product. These baselines will support the requirements for project documentation and software integrity through the software development life cycle.

5.1.1 Baseline Security

Baseline libraries will be created on both the mainframe and the LAN. Items included in the baseline will not be removed from their native environments during the creation of the baseline. All

mainframe items will be stored in secured libraries on the mainframe. LAN based items will be stored in secured directories on the LAN. The baseline libraries on both the mainframe and the LAN will be stored with protected status. They will be available to the project team for read access only. Copies of files may be made to work areas as necessary. Update or delete access to the baseline files will require change approval from the CCB.

5.1.2 Starting the process

CMT will submit a request for baseline creation to the CCB at the close of the project phases. The phases (Detailed Design, Construction and Testing, and Implementation) are terminated with project milestone events. When a request for baseline creation is received, the CCB will verify that the milestone for a phase has been approved. When approval is verified the CCB will direct the CMT to begin the creation of a new baseline.

5.1.3 Baseline Creation

The baseline is essentially a copy and record process. The following items will be copied from the project inventory in order to create the baseline:

- ❑ **Category I Items**
All approved software will be copied from the project working libraries into separate files on the central processing system. Copy processes for Category I software libraries will be carried out by the Technical Support Team Lead.
- ❑ **Category II Items**
All approved documentation and any supporting files will be copied into the baseline directories on the LAN. Copy processes for Category II files will be carried out by the Project Controller.
- ❑ **Category III Items**
Generally production data will not be copied as part of the baseline process, however production data used in testing will be part of the baseline process.

5.1.4 Baseline Completion

When a baseline has been completed by the CMT, the request will be completed. This will close the CI and notice will be sent to the entire project.

5.2 Archiving Processes

Archives are a secure record of project materials at a specific moment in time. They contain interrelated materials produced by the project such as documentation and software. The actual content of the archives planned for the WDTIP is described in the **Configuration Identification** section of this document.

At the close of the three phases of the project, coinciding with the creation of the project baselines, three archives will be created. The process outlined for the creation of a archive is

similar in content to that of the baselines. A planned archive is essentially an offsite, protected copy of the baseline. The security and creation process is described in the following sub-sections.

5.2.1 Archive Security

Archive libraries will be created on both the mainframe and the LAN. Items included in the archive will not be removed from their native environments during the creation of the archive. All mainframe items will be stored in secured libraries on the mainframe. LAN based items will be stored in secured directories on the LAN. The archive libraries representing both the mainframe and the LAN will be stored with protected status. They will be created on removable media and stored in an offsite location. Access to the information contained in the archive must be requested through the CMT. No update or modifications will be authorized to archive files. Responsibility for the archive process is defined in the **Configuration Security** section of this document.

5.2.2 Starting the process

CMT will submit a request for archive creation to the CCB at the close of the project phases. The phases (Detailed Design, Construction and Testing, and Implementation) are terminated with project milestone events. When a request for archive creation is received, the CCB will verify that the milestone for that phase has been approved. When approval is verified, the CCB will direct the CMT to begin the creation of a new archive.

5.2.3 Archive Creation

The archive is essentially a copy and record process. The following items will be copied from the project inventory in order to create the archive:

- ❑ **Category I Items**
All approved software will be copied from the project working libraries into separate files on the central processing system. The files will then be copied onto removable media for offsite storage in accordance with HSDC procedures. Copy processes for Category I software libraries will be carried out by the Technical Support Team Lead.
- ❑ **Category II Items**
All approved documentation and any supporting files will be copied into the archive directories on the LAN. The entire directory will then be copied onto removable media and stored offsite in accordance with procedures established by the LAN Support Group. Copy processes for Category II files will be carried out by the Project Controller.
- ❑ **Category III Items**
Production data will not be copied as part of the archive process.

5.2.4 Archive Completion

When an archive has been completed by the CMT, the request will be completed. This will close the CI and a notice will be sent to Project Management.

5.3 Configuration Control Process

The Configuration Control Process is the method by which a change to a CI is evaluated, approved, or disapproved, scheduled, and tracked after formal establishment of the CI. Configuration control is applied to ensure the consequences of change are taken fully into account.

Changes affecting the baseline characteristics of Category I, II, and III CI must be submitted for action through the change control process. Change control should only allow a change if a justification, impact analysis, and approval have taken place. Uncontrolled change of baseline characteristics, no matter how slight, will allow components to drift apart and eventually cause a serious incompatibility between the original and the derivatives.

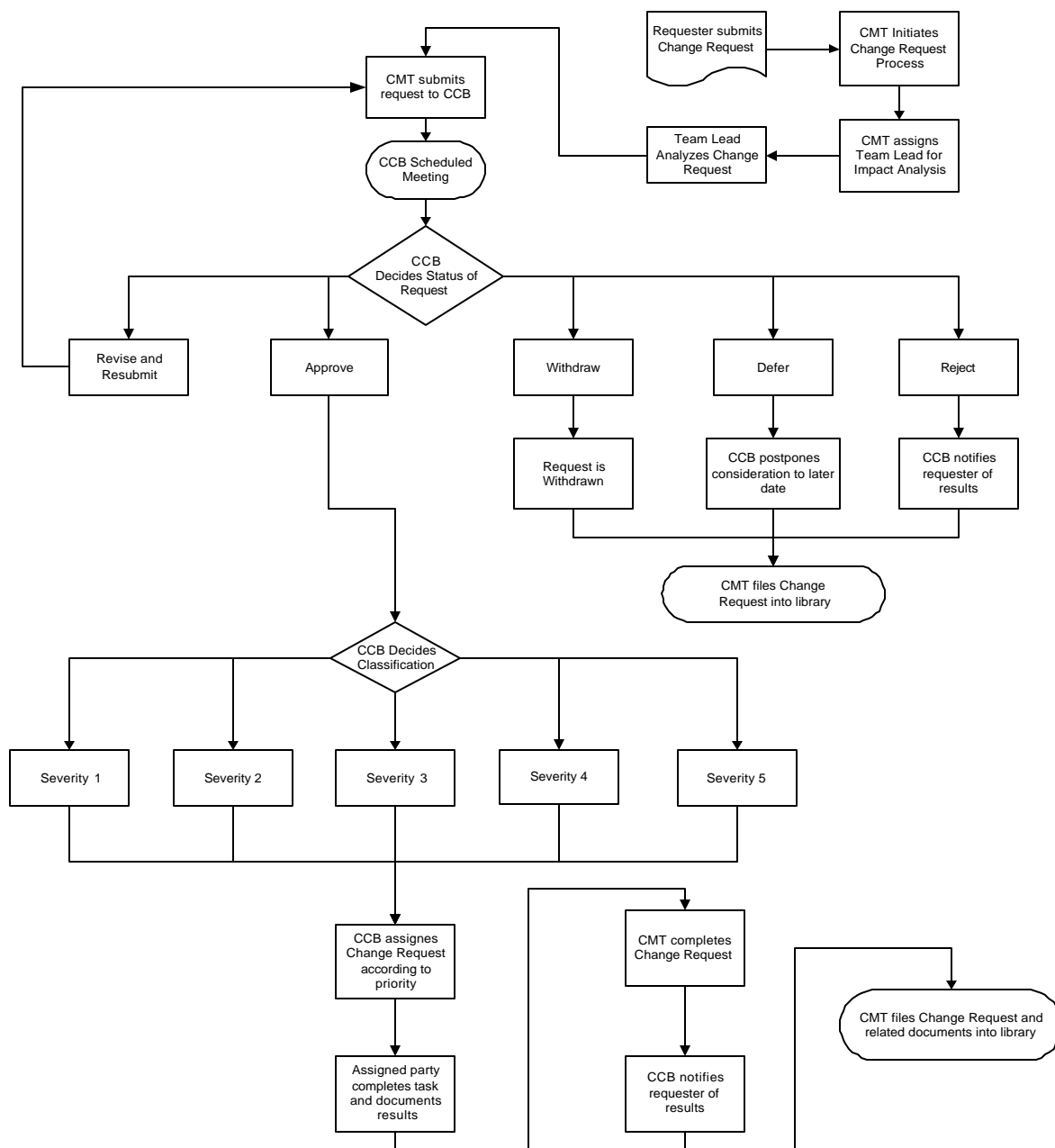
Category IV and V items will not be directly controlled by the CM process. Changes to these items that are reported to the project will be tracked as they relate to the controlled items in the first three categories. An example of this type of change would be a new release of a programming language used within the project. A change request would be used to analyze the impact of the release on the programming effort of the project. This analysis will allow the CCB to decide on the implementation of the release in the manner that has the least disruptive impact on the project.

5.3.1 Configuration Control Process Flow

A Configuration Item Change Request is the primary instrument for initiating a change to an existing requirement or feature of the system. Change request forms have been developed to support several specific functions as well as generalized requests. Examples of the change request forms are included in the Forms Appendix to this document. A change request may be initiated by the users of the TRAC Application, representatives of the client organization, or members of the project team.

This process is further documented in the figure on the following page.

Figure 4: Configuration Control Process Flow



5.3.2 Change Request Submission

Changes may be requested by either the users of the TRAC Application, members of the client organization, or by members of the project team. When the requirement for a change is identified a request for change is sent to the CMT. The CMT will assign a project Team Lead to review the change request and perform an impact analysis of the change. The Team Lead will complete the change request form and return it to CMT for inclusion on the agenda of an upcoming meeting of the CCB. The team lead will notify the original change requestor of the date on which their change request will be considered by the CCB.

5.3.3 Configuration Control Board Processes

All proposed changes will be submitted using the Project Tracking System. The tracking system will allow the assignment of configuration identifiers to the change requests automatically. Progress on a request will be entered into the system to support the tracking of change requests. A description of the Project Tracking System is included in the **Tools** section of this document.

The Team Lead assigned to analyze the change request will present it at the scheduled meeting of the CCB. After reviewing the change request, the CCB will determine if the change is approved, rejected, deferred, withdrawn, or requested to revise and resubmit. If the change request is approved, the Board will then determine a severity of the change and delegate a CCB member to manage the implementation of the change. Notice of the status of change requests will be communicated to the project through publication of the minutes of the CCB meeting.

The frequency of CCB meetings will vary at different phases of the project. The CM will determine the frequency of the meetings and schedule them according to the volume of change requests that are submitted during the week. The initial planned frequency of the meetings is biweekly.

5.3.3.1 Definition of Statuses for Change Request

After viewing the change request form, the CCB will then decide the status of the change request. There are five statuses of change requests:

- **Approve**

The change request is approved as is and no further changes are necessary to the change request portion of the form. The CCB members will then determine the classification of the change. Classifications of changes are described in the next section. The CCB will also assign a member to manage the progress of the change from approval through completion. The assigned change manager will report the status of the change request at future meetings of the CCB until the request is closed.

- **Reject**

The CCB has determined that the change request is found invalid or out of scope. If a change is rejected by the Board, the reasons for denial of the change will be documented and the requester will be notified by the member of the CCB who originally submitted the request for consideration.

- ❑ **Revise and Resubmit**
The change request must be revised and resubmitted to the Board. Further analysis needs to be done before resubmitting the request.
- ❑ **Defer**
The change request is deferred until a later date, possibly later in the life cycle of the project, or until resolution is possible.
- ❑ **Withdraw**
The change request is withdrawn by the requester or CCB. If a change request is withdrawn, no action is taken on the request.

All change requests and subsequent documentation (e.g., minutes of the meetings or approval/denial forms) will be recorded in the minutes of the meetings of the Change Control Board. Approved changes are included in the Change Control Log.

5.3.3.2 Definition of Priority Levels for Change Requests

If the change request is approved, the Board will then determine a classification of the change. These classifications characterize the relative impact of the change upon the TRAC Application system being written and the resources of the WDTIP. For approved requests, there are five classifications of changes. The determining criteria are shown in the following table:

Table 3: Definition of Priority Levels for Change Requests

Severity	Applies if a problem could:
1	a. Prevent the accomplishment of a project goal or system function b. Require resources that would jeopardize the remainder of the project
2	a. Adversely affect the accomplishment of a project goal or system function and no work-around solution is known b. Adversely impact the resources of the project and no temporary solution is possible
3	a. Adversely affect the accomplishment of a project goal or system function but a work-around solution is known b. Adversely impact the resources of the project but a temporary solution is possible
4	a. Result in a functional inconvenience or annoyance but does not affect a required project goal or system function b. Result in inconvenience or annoyance for project personnel, but does not prevent the accomplishment of those responsibilities
5	Any other effect

5.3.4 Quick Fix Process

Changes that impact limited functions and project resources may not require the consideration of the full CCB. The volume of small changes could burden the group to the point that significant change requests would not receive adequate consideration. The CMT lead may delegate or assign responsibility for authorizing changes in severity 4 and 5 to the Project Team Leads with the approval of the Project Management.

Changes that are performed in this manner will be entered into the Project Tracking System by the CMT lead or their designates. These items will then be tracked in the same manner as all other changes.

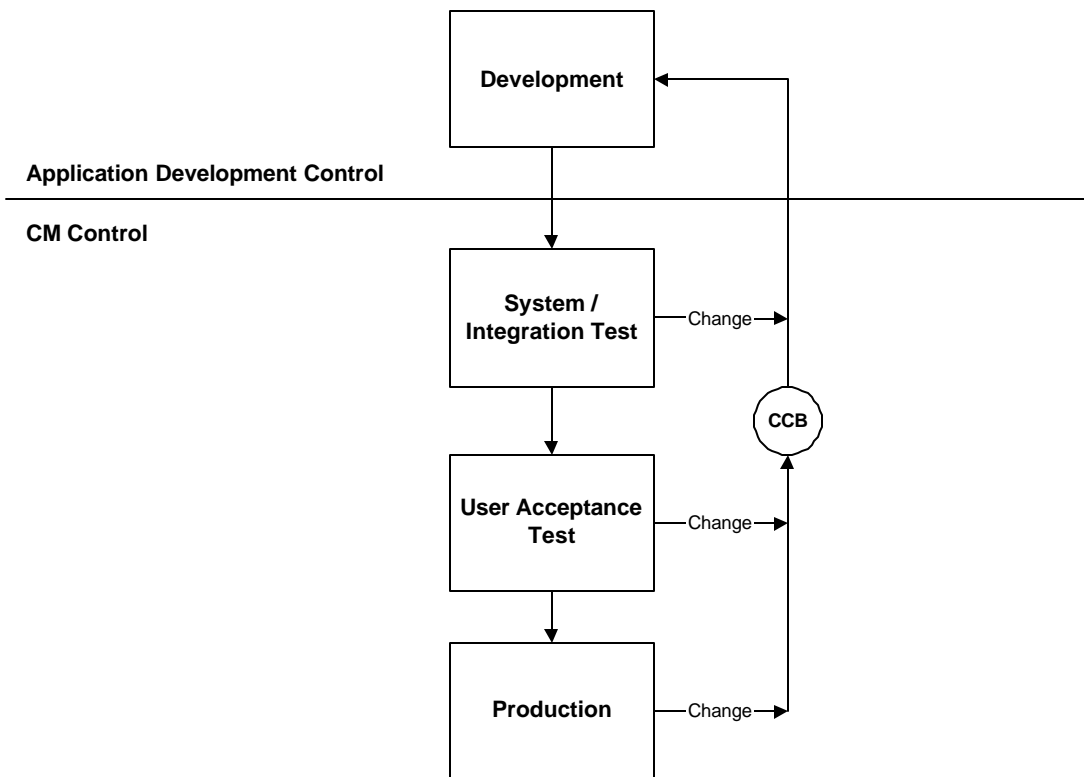
5.4 Software Control Process

The Software Control Process is the method by which a Category I item moves through the Software Development Life Cycle. This process is designed to prevent untested functionality from reaching the Product Baseline. This section will discuss the steps that Category I items must complete to become part of the Product Baseline.

5.4.1 Promotion Model

The central element of the Software Control Process is the Promotion Model. The model helps in the control of software as it is promoted from development to final release. It also shows graphically the point at which Category I items move from the development environment into the CM controlled environments. The WDTIP promotion model is shown in the following figure.

Figure 5: WDTIP Promotion Model



The default promotion path a software item takes during the system development life cycle is to move in a straight line from the Development region, through System / Integration Test, User Acceptance Test, and on to Production.

When changes are required in software items they are demoted to Development to be modified. The move to Development takes place without regard to the environment that the item resided in when the change was defined. When changes are completed, items are promoted to the appropriate testing stage for re-testing. This promotion and demotion cycle continues until all testing is completed.

Backups are taken at each level of the promotion process to insure the integrity of the changed environment. At no time will a Category I item replace a controlled environment item without a backup of the existing item first being taken. The process for assuring this level of backup and recoverability is fulfilled through the tools used for the migration process.

5.4.2 Starting the Process

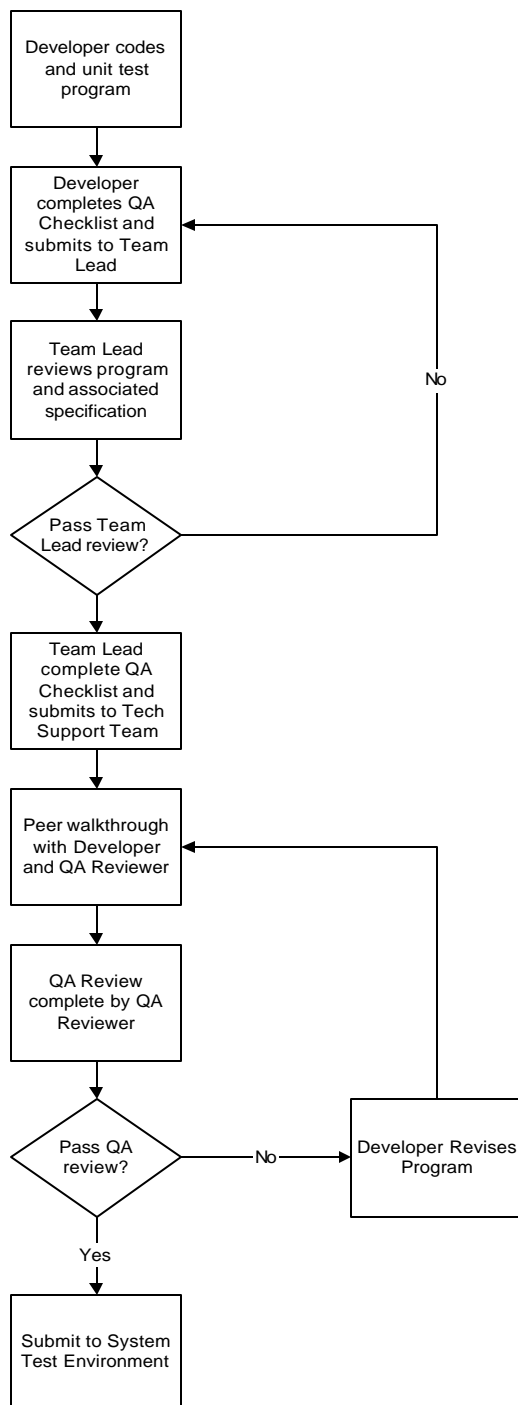
Programs are initiated in the Development Environment. During the Development Phase of the WDTIP, developers will be primarily responsible for the initial programming of the each of the program specification created during the Design Phase. As the programs are completed and prepared for System Test, a quality assurance (QA) process has been established to help insure the program:

- ❑ Meets the functionality identified in the program specification
- ❑ Uses code that is both efficient and correct, and
- ❑ Meets the standards established by the Design and Coding Standards

The specific steps of the QA process that will be used during Unit Test are identified below.

1. **Program Submission-** Developer completely finishes coding program and documents any changes necessary to the specifications. The developer should also run a query to identify all copybooks and other programs referenced by the program. The developer should personally complete the QA checklist, completing the Complete or N/A. Once the developer feels the program is ready to be moved into the System Test Environment, he/she should submit the QA Checklist to the Team Lead for a review of the program functionality and team requirements.
2. **Team Lead Review-** The Team Lead will review the QA Checklist and program to insure all of the necessary changes have been made to specifications and the data model and the unit test has been successfully completed. Once that has been completed, the Team Lead should sign-off on the QA Checklist and forward it onto the Technical Support Team for the QA review.
3. **Peer Walkthrough-** Once the QA Checklist has been submitted to the Technical Support Team; a walkthrough of the code should be completed by the Developer and QA Reviewer. This should include a walkthrough of the source code, a demonstration of the functionality and/or the output files for that program.
4. **QA Review-** After the developer has conducted a walkthrough of the code and functionality, the QA Reviewer will complete the QA checklist independent of the Developer. This checklist includes individually testing the functionality of the program utilizing the Unit Test Bed Data. If any components of the checklist do not pass this review, the program will be returned to the programmer for revisions. Comments will be provided in the QA Comments Area of the QA Checklist if the correction or modification is not clear.
5. **Revise Program-** This step involves the programmer modifying the program based on the comments of the QA Checklist. Once all of the comments have been completed, the Developer will resubmit the program to the QA Reviewer for approval. Again if any of the components of the checklist are not satisfactorily met, the program will be resubmitted to the Developer for modification and correction.
6. **Unit Test QA Approval-** After the program is revised it will be prepared for migration to the System Test Environment. All associated program and copybooks shall be identified by the Developer to accompany the program in its migration process.

The flow chart on the following page depicts the steps of the QA Process.



5.4.3 Software Migration Process

CMT will review and process software promotion requests. Move Request forms will only be processed that have the approval of the appropriate persons. Three levels of migration are possible:

- ❑ **Development to System / Integration Test Migration**
Move Requests to the System / Integration Test Environment will require the authorization of the Application Team Lead. Supporting documentation shall be attached verifying that a Quality Assurance Review was completed and approved by the QA Analyst. Approval from the Test Team Lead will also be required to migrate the new software into the System / Integration Test environment.
- ❑ **System / Integration Test to User Acceptance Test Migration**
Move Requests to the User Acceptance Test Environment will require the authorization of the User Acceptance Test Team Lead. Supporting documentation will be attached verifying that System Testing was completed and approved by the System Test Team Lead and QA Analyst.
- ❑ **User Acceptance Test to Production Migration**
Move Requests to the Production Environment will require the authorization of the Implementation Coordinator. Supporting documentation will be attached verifying that User Acceptance Testing was completed and approved by the User Acceptance Test Team Lead and QA Analyst.

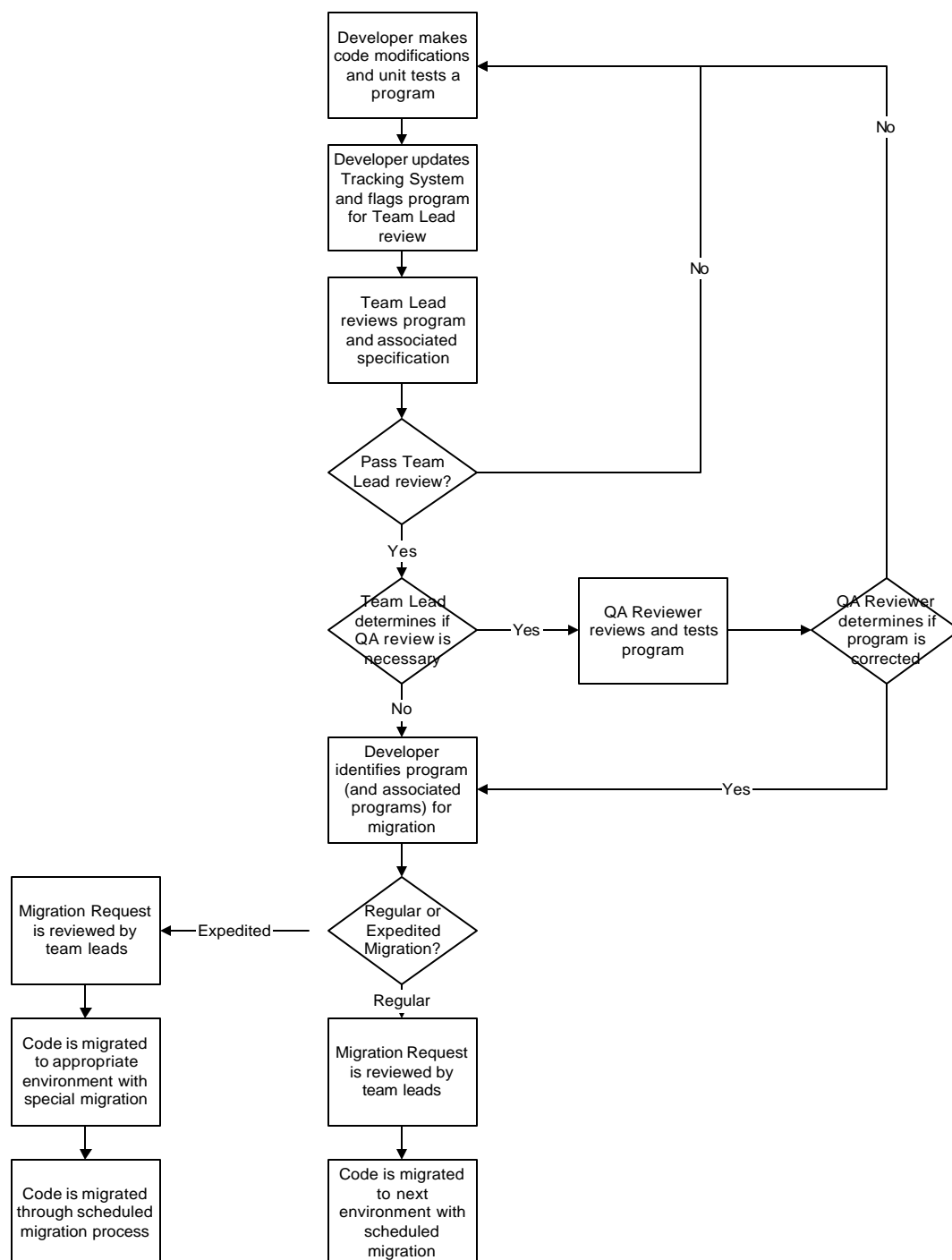
The process of software migration and tracking it through CM will begin after the completion of the initial code and unit test of all of the programs. Once that migration has occurred, there will be a separate QA process for the migration of the programs described above.

After the programs have been moved into the System Test Environment, the QA process will utilize the Project Tracking System developed in Access. This database is designed to allow tester and users of the system to document any problems encountered in the system and track the resolution of those problems by the application team. Program Change Requests (PCR) will be assigned to a specific developer who will make modifications to the code to correct any programming errors. Once the developer has completed the corrections to the code and tested the program, the QA process will begin before the code can be migrated to the System Test Environment (and beyond). Below is a description of the steps in the QA process.

1. **Team Lead Review-** The programmer will complete the program and check the updated code back into the Development Reference Library (described in the Library Management Design Section of the Configuration Management Plan. The Team Lead will then review the PCR, and check if the identified problem has been corrected. The Team Lead will then decide if the problem has been corrected. The Team Lead will also decided that if the program is very complex, has SQL changes or has impacts across many program, to have an additional QA of the program by the Technical Support Team. This type of review may also be requested by the System Test Lead or User Acceptance Test Lead.

2. **QA Review (Optional)**- The QA Reviewer will review the code and test the functionality of the program if a request is made by the Team Lead, System Test Lead or User Acceptance Test Lead. This review is primarily to review complex changes the SQL or COBOL code. This review can also be design to check the performance of a program if substantial changes have been made or it is a critical program. Once the QA Review is complete, the QA Reviewer will either return the code to the developer for more modifications, or OK the Code for Migration.
3. **Migration of Code**- Once the program has been reviewed and it is ready to move to the next environment, the developer will complete add program (and any impacted JCL's, programs, DBRM's, copybooks, and JCL Control Cards) to the Migration Request Form. This form will be maintained in the Project Tracking System and administered by the Technical Support Team. Once all of the developers have identified the appropriate code for migration, the Technical Support Team will circulate the completed list to the Team Leads for Review. The Team Leads will then provide approval for the migration of data to the next environment. This migration will be completed on a regular schedule (weekly) but can be adjusted to meet the needs of the project.
4. **Expedited Code Migration**- In some instances a critical PCR may be discovered that cannot wait for the regular migration schedule. An example of this could error in a program that is necessary to provide data to the training environment discovered a few days prior to a training session. When this occurs, a separate migration request can be made to only include the programs associated with the critical PCR. This PCR will still need to go through the necessary QA process (i.e., the Team Lead or QA Reviewer), but then may be able to be migrated directly to the environment where the bug was discovered. Once the correction has been migrated to the appropriate environment, it will be the responsibility of the Team Lead to insure the code gets incorporated into any concurrent code development that could be occurring during this migration. This type of migration will need Project Management approval.

The follow flow chart depicts this process.



5.4.4 Migration Completion

When a Category I item migration has been completed by Technical Support, the request will be completed. CM will return the completed Move Request to the originator as formal notice that the move has been successfully performed.

5.4.5 Processing Rejected Items

Category I Items that are rejected as a result of failed compiles or tests will be sent back to the originating group for investigation. The Technical Support team will provide supporting documentation and data to aid in problem resolution.

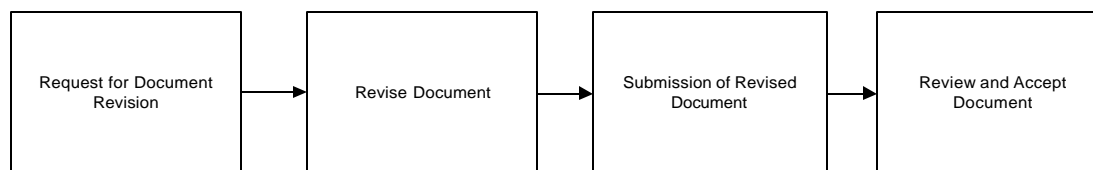
5.5 Document Control Process

The Document Control Process is the method by which a Category II item moves through the revision process. The revision process within the context of this section will describe revisions that are made to the document after it has been officially approved. This section will discuss the steps that items must complete in this revision process.

5.5.1 Document Control Process Model

This model illustrates the flow of Category II items from the development stage to the approval stage during the revision process.

Figure 6: WDTIP Document Control Process Model



5.5.2 Document Control Stages

The following are the control stages for Category II items:

- ❑ **Request for Document Revision**
All requests for document revisions will be made using the same configuration control process outlined in the **Configuration Control Process** section of this document. In addition, individuals tasked with reviewing the document once it has been revised will be identified at the same time that the change request task is assigned.
- ❑ **Revise Document**
Since all approved documents cannot be edited from the “approved” folder on the WDTIP directory structure, the PC will make a copy of the approved document, increase the version number by one for the copied version to be revised, and transfer it to the “development” folder. All team members tasked with revising the document

will have access to in-process documentation that is filed in the “development” folder stored on the network.

❑ **Submission of Revised Document**

When the revised document is ready to be submitted to the State, the team will forward the document’s path to the Project Controller (PC). The PC will store the document in the “submitted” folder on the network. This is necessary, as the PC is the only individual with write access to the “submitted” folder. The PC will email the State indicating that the document has been stored in the “submitted” folder and is ready for review.

❑ **Review and Accept Document**

The individuals tasked with reviewing the document will know that the document has been revised and is ready for review once they have received the email notification the document has been transferred to the “submission” folder. Once the document is approved and the PC has received notification from the State regarding the approval, he will then move the deliverable document to the “approved” folder.

When a managed document is changed, a new version shall be created. The file name and version number combination will be recorded in the completion information on the change request that initiated the process. When the completed change request is recorded in the library, the version number will be included in the recorded data. The PC is the only individual with write access to the document once it is placed in the “submitted” folder. The PC will increase the version number by one once the document revision has been requested and the document to be revised is moved from the “approved” folder to the “development” folder for revision. At that time, the document is updated and resubmitted to the State after being revised.

5.5.3 Web Site Controls for Documentation

All documents to be posted to the WDTIP web site will come directly to the Project Controller. The PC will determine that the document version is the most current and that the document is located within the Approved folder on the LAN (local area network). The PC will communicate directly with the WDTIP webmaster regarding the location of the document to be posted to the web site. At that point, the WDTIP webmaster will initiate the State’s approval process. Upon the State’s approval, the webmaster will post the document to the web site.

5.6 Data Control Process

The Data Control Process is the method by which Category III items are managed. Production data is not developed within the project, and therefore promotion processes are not pertinent to this category of items. Change requests concerning data originate primarily from errors and omissions in the data stream from the source systems to the central database of the project. Changes may also be requested in the central database due to changes in, or additions to, the requirements for the system. Examples of each type of occurrence, and the process to deal with it follow:

❑ **Production Input Data**

Input data files are received by the WDTIP on a daily basis. They conform to a fixed file format and are processed according to that specification. The application programs of the system perform a final review of the data they are processing before inserting it into the central database.

If errors are found, those records are not loaded to the database. They are moved to a rejected records file. Analysis is performed by the project to locate the reason for rejecting the records. The results of this analysis will be communicated with the originator of the file and a course of action will be decided on.

If a correction of the rejected records is required, it will be performed by the Maintenance and Operations staff of the implemented system. The file will then be reprocessed. If correction and resubmission of the file is required, that work will be performed at the originating organization.

❑ **Central Database**

The central DB2 Database of the project will contain the sum of the information processed from all of the counties. In addition to this transactional information, several reference tables will be required for the processing of coded information in both the batch and online environments.

If a requirement for a new coded entry is uncovered after implementation, a request to add that entry to the production tables will be submitted to the Maintenance and Operations Team. They will perform the change and report the status of change back to the requesting organization.

Processes for dealing with production data files and databases refer to organizations that will not come into existence until the close of the project. Refer to the Transition Plan of the project for full information on these processes.

5.7 Incident Tracking Process

Incident Tracking Management is a process used to report, track, and assign resolutions to anomalies found within the TRAC Application during integration/system and user acceptance testing phases. The definition of what constitutes an incident is described in the Definitions and Acronyms of this document. Incidents that arise during testing need to be tracked, categorized, classified and tasked to the appropriate parties at any time during the SDLC.

When incidents arise, they need to be resolved in a consistent and disciplined manner in order to maintain the quality of the deliverable, as well as to control schedules and cost. The described incident tracking process ensures that incidents are defined properly, escalated for management attention, and resolved quickly and efficiently.

The incident escalation process consists of five steps. These steps are:

1. Incident identification
2. Complete and submit an incident form
3. Incident review and assignment
4. Develop and execute resolution of incident
5. Incorporate incident resolution

❑ **Incident Identification**

Project incidents may originate from sources within the project, or from outside sources, such as end users, policy specialists, technical staff, etc. Traditionally, most incidents originate with either the project team members or the end-user.

❑ **Complete and submit an incident form**

The Incident Tracking Module within the Project Tracking System(PTS) is used to identify and report incidents (Refer to the **Tools** section of this document). Each incident refers back to a program and has a unique identifier used to track the incident. The person who enters the incident into the PTS is called the submitter.

❑ **Incident review and assignment**

New incidents are reviewed by Team Leads who then decide the status and classification of each incident. Incidents are prioritized by urgency, then classified and assigned to the appropriate analyst. Incident prioritization and classification is accomplished using the Incident Tracking Module of the PTS. In addition, team leads are responsible for reporting all incidents to the CCB.

❑ **Develop and Execute Resolution**

It is the responsibility of the team leads to assign and follow up on incident resolution. Once the tasks have been completed to correct the incident, the person responsible for completing the incident must document the steps taken for resolution in the PTS.

❑ **Incorporate resolution into Project Tracking System**

Depending upon the incident and the classification of the incident, the WDTIP team members or the appropriate parties will be notified of resolution via the email system. It is the responsibility of the team leads to notify all parties that will be affected by resolution of the incident.

5.8 Configuration Security

At the core of Configuration Management is security. Each process and activity described in this document is concerned with distinctions between secured and non-secured information. Two broad areas of activity are discussed involving security. Unsecured items may be migrated into secured environments, thus changing their status into secured items. Changes may be required in items already approved in secured environments, causing multiple versions to be created.

The ultimate authority to make a change to the configuration of the project rests in the CCB. Individual configuration activities are carried out by specific individuals within the project. These individuals will have the necessary authority and access to perform the activities they are tasked to do. The individual roles and responsibilities of project members as they relate to configuration security are shown in the following table.

Activity	Responsible Party(s)
Baseline creation	Technical Support Group Team Lead Project Controller
Archive creation	Technical Support Group Team Lead Project Controller
Category I Item Migration	Technical Support Group Team Lead
Category II Item Migration	Project Controller
Category III Item Migration	Technical Support Group Team Lead
Version Control	See the underlying activity that caused the version to be required for the responsible party.
Mainframe Security Administration	Technical Support Group Team Lead
LAN Security Administration	Project Controller

6 Configuration Status Accounting

Status Accounting is the process of recording, documenting, and reporting the current status of all CI. The CMT will ensure that all logs, reports, and version description documents are maintained.

6.1 Logs

Logs are used to record all activities that impact a CI. They provide a historical record of the sequence of events that took place from the point of initial release. The logs are designed to:

- ☐ Record all activity that impacts the product
- ☐ Create a historical record of each CI development
- ☐ List and describe all problem reports, modification records, change notices and change requests

6.2 Reports

Reports are used to inform development team members and external parties of the status of a CI. They provide summary information of the status of all change activities during a particular reporting period. The reports are designed to provide information pertaining to:

- ☐ Status of change requests
- ☐ Problems detected from the time testing commences through software release
- ☐ Status of action taken on problems identified
- ☐ Status of a deliverable's progression

6.3 Accounting Process

The Project Tracking System (Refer to the **Tools** section of this document) will be utilized for logging all categories of CIs. The tool is an online database that will record and track relevant information regarding each CI incident in a summarized format to facilitate the incident resolution process. The Tool can generate reports regarding a CI.

Category II logs are effectively maintained by the Project Controller using a combination of document version numbering and email correspondence with the State regarding each document's status (approval/disapproval) through the approval process. The Project Controller can generate reports upon request that provide information pertaining to the document's progression through the approval process.

7 Configuration Auditing

Auditing refers to the review of the project team's work to assess the work's quality and compliance with project standards. IV&V, internal QA, and external QA will do auditing of CIs. The frequency and process by which IV&V will audit project work is described in the **IV&V Plan**. Refer to the **Project Management Plan** deliverable for further details about the role of the internal and external QA.

8 Configuration Tools

Tools listed in this section include a combination of custom applications and third-party software purchased and used within the project to aid in the CM activities. This section of the document does not seek to describe all of the operating features of a tool, but only those aspects that are significant to CM functions. This is not an exhaustive list of tools utilized in the project. Rather, the list only contains those tools utilized in the CM activities.

8.1 LAN (PC-Based) Tools

8.1.1 Project Tracking System (PTS)

The project team will develop an online PTS to record and track change requests of configuration items. The PTS is a custom application that will have various tracking modules that will record different types of change requests. Below is a list of tracking modules that the PTS will support:

❑ **Incident Tracking**

The Incident Tracking module shall be developed to record and track relevant information regarding Category I incidents. The *primary* information maintained in the Incident Tracking module includes the incident number (unique identifier), incident priority, incident description, current status of the incident, assigned to, and resolution description. Other pertinent fields integral to the facilitation of incident tracking will be incorporated.

The Incident Tracking module will have the ability to create management reports. For example, the Incident Tracking module can provide a report on all new incidents. An example of what this report might look like is shown below:

Detailed Report of Outstanding Incidents					
Date MM/DD/YYYY					
#	INCIDENT DESC.	PRIORITY	STATUS	STATUS DATE	ASSIGNED TO
100	B3IN screen locks up upon enter.	High	Open	01/26/95	Tummala, Kishore

❑ **Other PTS Modules**

The project team may develop other PTS modules in order to facilitate the CM change request process. The project team will review the various forms that are used within CM (see **Forms** section within the **Appendices** section of this document) to determine whether they can be incorporated into PTS as separate modules.

8.2 Mainframe Tools

8.2.1 Library Management Tool

Library management is the process of manipulating the contents of source libraries in a controlled and protected manner. The goal of library management is to provide a controlled and protected environment for the development and testing of program source code and the related objects that support that code.

8.2.1.1 Terminology

In the course of this **Configuration Management Plan**, several specific utility programs and other computing terms will be used frequently. The following glossary is offered as an aid in understanding these terms:

Backup – A process of creating a copy of an item to prevent the loss of work.

Generation Datasets – Generation Dataset Groups (GDG) are defined storage on the mainframe designed to provide multiple versions of the same structure. GDG base entries are defined to name the dataset and provide the characteristics and maximum number of versions allowed. When a new version of the dataset is created, the system assigns a generation number to the dataset. When the maximum number of versions is reached, the oldest generation is deleted each time a new generation is created.

IEBCOPY – The Library Copy Program is designed to copy libraries, or the members of libraries, from one location to another.

IEBGENER – The Sequential Copy/Generate Data Set Program is designed to copy datasets from one location to another and to create, or generate, new datasets.

Library – A collection of related items stored in a defined, and named, structure.

PDS – The Partitioned Data Set (PDS) is a storage format in mainframe systems. The contents of the data set are grouped into named collections called Members. The Members of a PDS may be referred to independently of each other in mainframe functions and within both batch and online processes.

RACF – The Resource Access Control Facility (RACF) is an application provided by IBM to define and enforce access security for mainframe resources such as datasets and programs.

8.2.1.2 Source Library Management

Source Libraries contain the program code and related software that comprise the operating components of the application. Source Libraries are defined on the mainframe as Partitioned Data Sets (PDS). The following libraries have been specified in the **Design/Coding Standards Guide**:

Library Type	Library Name	Uses
COBOL Source	B3.xxxx.SRCLIB	Source programs
COPYBOOKS	B3.xxxx.COPYLIB	Copybooks / CICS maps
JCL	B3.xxxx.JCLLIB	Job statements
JCL Procs	B3.xxxx.PROCLIB	JCL Procedure library
Control Cards	B3.xxxx.CNTLLIB	JCL Control card library
PROGRAM LOADs	B3.xxxx.LOADLIB B3.xxxx.CICS.LOADLIB	(Batch Program Load Modules) (Online Program Load Modules)
TABLE Copybooks	B3.xxxx.DCLGEN	Declaration generator generated copybooks for DB2 tables

In the Library Name, the variable in the second node denotes the libraries' environment. It is replaced by one of the following values:

DEV	Development/Unit Test
TEST	System/Integration Test
ACC	User Acceptance Test
TRNG	Training
PERF	Performance
PROD	Production

Each of these libraries will provide a storage location for a reference set of modules for an environment. Reference libraries are used within an environment as a common repository of software.

In addition to these reference libraries, a set of working libraries will be defined in the development environment. The working libraries will be the location of application development activity for all module types. Modules located in the working libraries are highly dynamic and may be incomplete or untested. Development library management deals with the movement of modules between the reference and working libraries. Working libraries will be named using the following standard:

Library Type	Library Name	Uses / Notes
COBOL Source	B3.DEV.WORK.SRCLIB	Source programs
COPYBOOKS	B3.DEV.WORK.COPYLIB	Copybooks / CICS maps
JCL	B3.DEV.WORK.JCLLIB	Job statements
JCL Procs	B3.DEV.WORK.PROCLIB	JCL Procedure library
Control Cards	B3.DEV.WORK.CNTLLIB	JCL Control card library
PROGRAM LOADs	B3.DEV.WORK.LOADLIB	(Batch Program Load Modules)

Library Type	Library Name	Uses / Notes
	B3. DEV.WORK.CICS.LOADLIB	(Online Program Load Modules)
TABLE Copybooks	B3. DEV.WORK.DCLGEN	Declaration generator generated copybooks for DB2 tables

The process of managing the project source libraries is divided into several functions. The following table shows the functions and the library types that they apply to.

Function	Description	Library Type
Module Check-Out	Move source modules from reference libraries into development working libraries	All except LOAD. New working LOAD modules are created from compiles.
Module Check-In	Move source modules from development working libraries into development reference libraries	All except LOAD. Multiple versions of LOAD modules are not required
Module Promotion	Move modules from reference libraries of any environment into the reference libraries of another	All
Schedules Backup	Copying libraries in any environment to protect them from loss of work	All

Module Development Tracking

Module development tracking is designed to record the status of all modules currently under development. A written log of all modules within the application is maintained by the technical support group to track the activity in both the working and reference libraries. A table named SIS_LIBRARY_AUDIT will be maintained in DATABASE to record and track the progress of modules as they are moved within the development environment.

There are two audit programs (B3DADT01 & B3DADT02) which will do the validation and insertion in the DATABASE. The following fields will be inserted in the table depending on the operation performed (CHECK-OUT or CHECK-IN or PROMOTION):

MODULE NAME
MODULE TYPE
OPERATION PERFORMED
RESOURCE ASSIGNED
CHECK-IN DATE
CHECK-IN SOURCE LIBRARY
CHECK-IN DESTINATION LIBRARY
CHECK-OUT DATE
CHECK-OUT SOURCE LIBRARY
CHECK-OUT DESTINATION LIBRARY

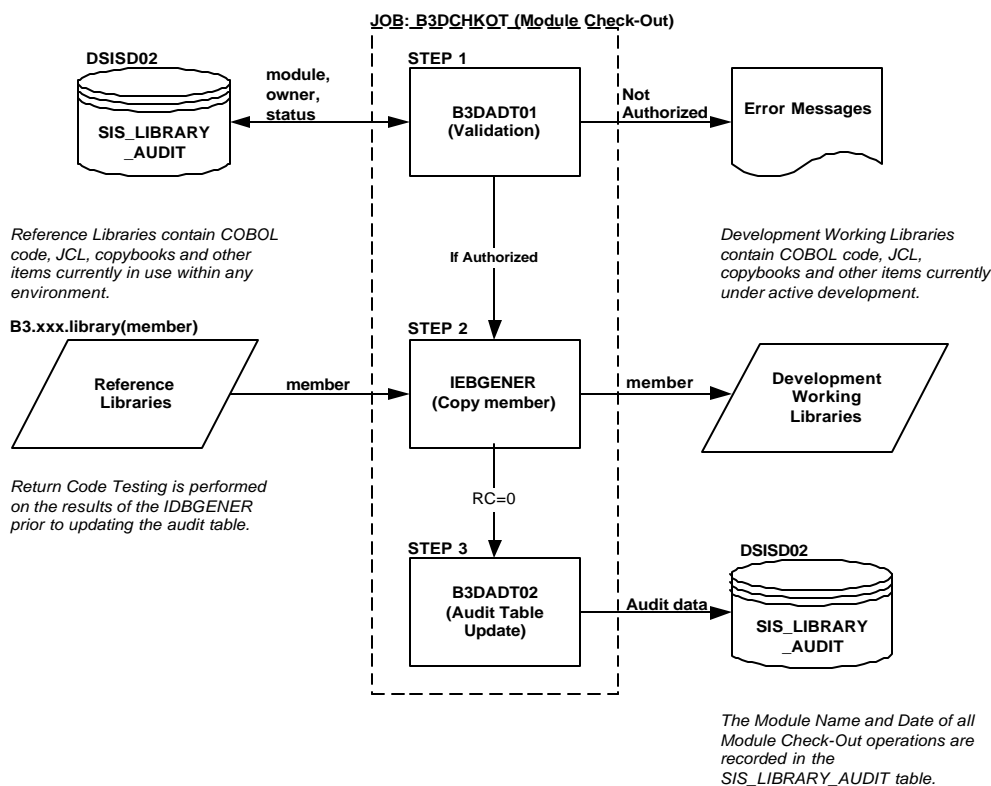
PROMOTION DATE
PROMOTION SOURCE LIBRARY
PROMOTION DESTINATION LIBRARY

Module Check-Out

When an existing module is assigned for development work, it must be copied from a reference library into a working library. The reference library source will vary with the requirement that caused the work request. New development may use an existing module with similar characteristics as a template for a new module. Change requests will require that a module be returned from its current environment into the development working library for alteration.

Logic in the Check-Out software will protect an existing module from being replaced with a new copy from the reference library. The following flow chart depicts the process of copying a reference module into a development environment working library.

Figure 7: Reference Module Copying Process



The following items explain the Check-Out process further.

- ❑ **SIS_LIBRARY_AUDIT:** A DB2 table containing the history of module check-out activity within the project.

- ❑ *Step 1: B3DADT01 (Validation):* A custom program that tests the check-out request against the information contained in the library audit table to validate whether the request should be allowed.
- ❑ *Error Messages:* A displayed message indicating the reason for disallowing a check-out request.
- ❑ *Reference Libraries:* A member located within any of the defined libraries in any of the environments of the project.
Ex: B3.ACC.SRCLIB(B3DBC01)
- ❑ *Step 2: IEBCOPY (Module Check-Out):* The specified member of the specified reference library is copied, and a new member with the same name is created in the specified development working library. If an existing member with the same name is found in the working library, the copy will be terminated. A terminated copy will not overlay the existing member in the development working library that was the target of the requested copy.
- ❑ *Development Working Libraries:* A new member located within any of the defined working libraries in the development environments of the project.
Ex: B3.DEV.WORK.SRCLIB(B3DBC01)
- ❑ *Step 3: B3DADT02 (Audit Table Update):* A custom program that inserts audit data documenting the check-out request into the library audit table.

Module Check-Out Security

The check-out process is a secured process. Access control security is defined for the components and processes of module check-out. The risk of inadvertently overlaying a module is addressed through security options within the check-out software that prohibit the creation of a new member in a working library with the same name as an existing member. RACF definitions for libraries and processes are shown in the following table:

Resource	Users	Access
Reference Libraries	All team members	Read Only
	Technical Support Group	Update
Module Check Out Job	All team members	Read Only
	Technical Support Group	Execute
Development Working Libraries	All team members	Update

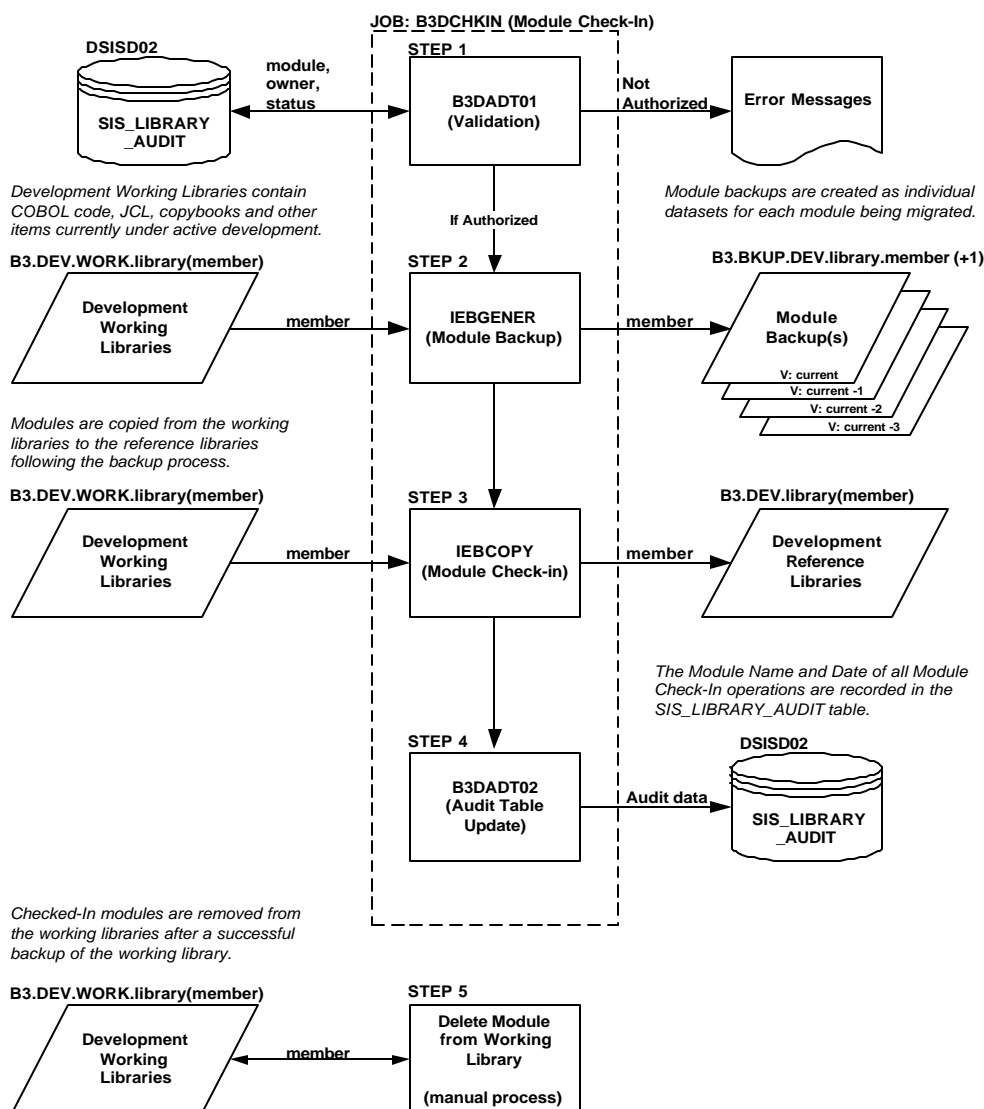
Module Check-In

When development work for a module is completed it must be copied from the development working library into the development reference library. The process used to accomplish this activity provides version control and backup functionality in addition to the transport function for the module.

The technical support group updates the written log of modules currently under development when the process is complete. See the example in the **Module Check-Out** section of this document for further detail on this process.

The following figure depicts the process of checking a finished module into a development environment reference library

Figure 8: Development Environment Reference Library Check-in Process



The following items explain the Check-In process further.

- ❑ *SIS_LIBRARY_AUDIT*: A DB2 table containing the history of module check-in activity within the project.
- ❑ *Step 1: B3DADT01 (Validation)*: A custom program that tests the check-in request against the information contained in the library audit table to validate whether the request should be allowed.
- ❑ *Error Messages*: A displayed message indicating the reason for disallowing a check-in request.
- ❑ *Development Working Libraries*: A member located within any of the defined working libraries in the development environments of the project.
Ex: B3.DEV.WORK.SRCLIB(B3DBC01)
- ❑ *Step 2: IEBGENER (Module Backup)*: The specified member of the specified working library is copied, and a new dataset is created to provide a saved record of the content of this version of the module. The new dataset is assigned the next sequential generation number available for that GDG series.
- ❑ *Module Backup(s)*: Datasets containing a series of versions of the module, arrayed from the most recent to the oldest. Module backups are stored in Generation Dataset Groups(GDGs)
Ex: B3.BKUP.DEV.SRCLIB.B3DBC01.G0001V00
- ❑ *Step 3: IEBCOPY (Module Check-In)*: The specified member of the specified working library is copied into a member of the same name in the specified development reference library. If a member exists in the reference library with the same name, it is replaced; otherwise a new member is created.
- ❑ *Reference Libraries*: A member located within any of the defined development libraries of the project.
Ex: B3.DEV.SRCLIB(B3DBC01)
- ❑ *Step 4: B3DADT02 (Audit Table Update)*: A custom program that inserts audit data documenting the check-in request into the library audit table.
- ❑ *Step 5: Delete Module from Working Library*: The specified member of the specified working library is manually deleted from the library.

Module Check-In Security

The check-in process is a secured process. Access control security is defined for the components and processes of module check-in through the RACF application. RACF definitions for libraries and processes are shown in the following table:

Resource	Users	Access
Reference Libraries	All team members	Read Only
	Technical Support Group	Update

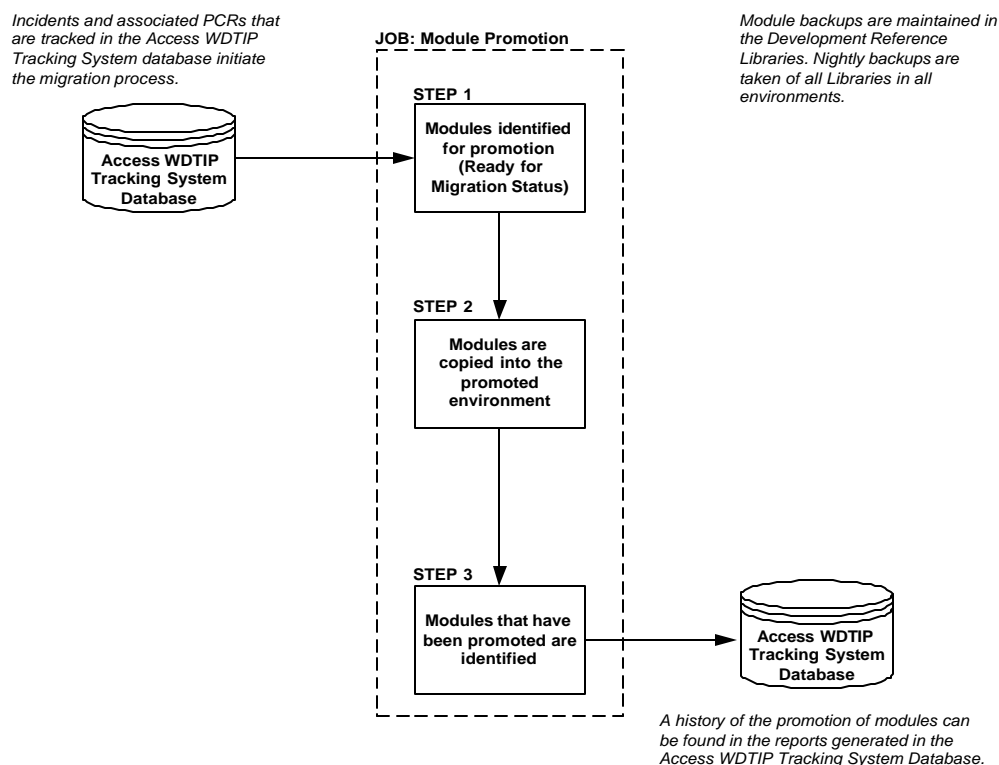
Resource	Users	Access
Module Check In Job	All team members	Read Only
	Technical Support Group	Execute
Development Working Libraries	All team members	Update

Module Promotion

Modules are promoted from one environment to another in accordance with the processes described in the *Software Migration Process* subsection of the *Configuration Control* section. Move requests are approved by the processes described in the plan and submitted to the technical support group for action. When a move request is received by the technical support group the module promotion process will be performed.

The following flow chart depicts the process of promoting a reference module into the reference library of a new environment. The flow chart is presented in a generalized form that applies to all environments.

Figure 9: Reference Module Promotion Process



The following items explain the module promotion process further.

- ❑ **Step 1:** As incidents are identified and the associated programs (PCR) are corrected, the incidents move to a status of “Ready for Migration” in the Incidents area of the Project Tracking System database. A report is generated displaying all of these PCRs and this report is used by a member of the Technical Support Team to migrate the appropriate code into the new environment.
- ❑ **Step 2:** The modules are then copied into the new environment. Once they have been migrated, analysis is completed by a member of the Technical Support Team to determine other programs that will be impacted by the recompilation of the migrated program. Then, all impacted programs are recompiled.

- ❑ *Step 3:* A Technical Support Team member will then update the Project Tracking System database by changing the status of the incidents.

Module Promotion Tracking

Module promotion tracking is designed to record the history and status of modules that have been promoted from the development environment into any of the controlled environments. The Audit Table Update program (B3DADT02) will be utilized to track version changes to any of the modules within the application. This program is maintained by the technical support group and is used to track the activity in the reference libraries.

A migration report is generated, for every migration request, which lists all the modules that need to be promoted to a designated environment as part of that particular migration. This report is generated from the PTS and utilizes the data associated to the actual incidents being migrated, to determine the actual modules that have been changed. This ensures accuracy of the report and helps improve the overall quality of the migration process since the list of objects changed is only kept in one place.

Module Promotion Security

The module promotion process is a secured process. Access control security is defined for the components and processes of module promotion through the RACF application. RACF definitions are shown in the following table:

Resource	Users	Access
Reference Libraries	All team members	Read Only
	Technical Support Group	Update
Module Promotion Job	All team members	Read Only
	Technical Support Group	Execute

8.2.1.3 Scheduled Backup

Scheduled backups of the reference libraries will be taken in accordance with the guidelines defined in the **Detailed Design Specification Document**. Detailed information on the schedule period and retention characteristics of the backup process is available in that document. Working libraries will be backed up in their entirety on a daily basis.

Specific member backups associated with the check-in and migration processes are not scheduled events. These copy operations are carried out as part of the larger process in order to provide additional security and versioning capabilities.

8.2.2 Platinum Database Utilities

Platinum Technology, Inc. has developed a suite of tools designed to aid database administrators in the management of DB2 database objects in a mainframe environment. These tools provide a sophisticated and flexible environment for the development and maintenance of DB2 databases.

The tool suite is composed of four primary applications and several related utilities. Two primary applications will be used extensively within the project.

RC/Query provides an ISPF styled interface for analysis of the structure of DB2 objects. By providing formatted viewing capability of the DB2 catalog, the details of structure of a database may be more easily analyzed.

RC/Migrate provides an automated method of modifying and migrating DB2 objects. Search capabilities are provided to allow objects to be grouped as necessary for the work required. Built in capabilities allow the database administrator to calculate physical parameters of the objects in a flexible manner.

Baseline, Archive, and Promotion activities are supported through RC/Migrate. The tool provides the capability to gather all related DB2 object definitions and their related criteria into a single file. That file can be stored in baseline library, moved to an archive for offline storage, or promoted to another environment as required.

The project uses these tools in the maintenance of database objects within environments. The tools are also used in the promotion of database objects to controlled environments and in the creation of baseline copies of the database definitions.

Detailed descriptions of the Platinum Database Utilities and their features are available in the product documentation. Both electronic and paper versions are available.

8.2.3 RACF

Security and access control are common requirements of many of the CM procedures. Mainframe security is managed using the Resource Access Control Facility (RACF) provided by IBM. RACF provides a flexible means of controlling access to mainframe datasets through both command level and ISPF interfaces. Control may be defined in broad classes or as narrowly as a single dataset.

Capabilities provided by the tool include the ability to define a class of users, grant access privileges to that class, and then connect individuals to the class. This allows access to be controlled at a group level where multiple individuals require similar access capabilities. The tool also allows the access profile of an individual to be tailored to the specific requirements of the position. This allows the creation of supervisory and special function users within the environment.

Access privileges may be created at the level of both the user and the resource being accessed. This capability will be used to protect the baseline and archive files after they are created in the project.

Detailed descriptions of the capabilities of RACF are available in the product documentation. Both electronic and paper versions are available.

9 Appendices

9.1 Forms

Forms are used within CM to support, organize, and document many of the configuration control processes. Examples of these forms are included in this section. Form samples are stored in H:\WDTIP\Forms. All team members will have access to CM forms from the project LAN.

The project team is currently reviewing these forms to determine whether or not they should be incorporated into PTS as separate modules. This will allow CM to more easily document and track the configuration control processes.

The CM forms include:

- ❑ **Configuration Item Change Request**
The Configuration Item Change Request form is a generic form that is initially completed to request changes to a configuration item. Depending on the categorization of the change request, other forms may need to be submitted such as the Data Model Change Request Form, the PCR, the Database Change Request Form, etc. If no additional forms are necessary, then the Configuration Item Change Request will be considered the only required form submission.
- ❑ **Data Model Change Request Form**
Requests for changes to the logical data model that do not effect the physical structure of the database should be made through this form.
- ❑ **Database Change Request Form**
Requests for changes to the physical structure of the database should be made through this form.
- ❑ **Quality Assurance Review**
The Quality Assurance Review form is completed prior to requesting the migration of a new, or substantially modified, program from the Development region into the System / Integration Test region.
- ❑ **Move Request**
The Move Request is completed to request the migration of WDTIP software to the controlled host environments (i.e. System / Integration Test, Acceptance Test and Production).

9.1.1 Configuration Item Change Request Form

Configuration Identifier: _____

Change: _____ Date: _____

Category: _____ (I though V)

Requestor: _____

Configuration Control Board Consideration: _____ Date: _____

Configuration Control Board Decision:

Approved _____ Date: _____

Deferred _____ Date: _____

Rescheduled?: _____ Yes/No _____ Date: _____

Rejected _____ Date: _____

Change Description:

Impact Analysis:

Review's Comments:

9.1.2 Data Model Change Request Form

Configuration Identifier: ____

Change: _____ Date: _____

Requestor: _____

Database Administrator Approval: _____ Date: _____

Change Description:

Impact Analysis:

Change Items	Completed

9.1.3 Database Change Request Form

Change Request ID: ____

Table Change ☐ **or** New Table ☐ **or** Other DBA Request ☐

Requested Implementation Date: _____ Requester: _____

Database Administrator Approval:

Date: _____

Other DBA Request Description:

Request Reason:

Table Name : _____

Table Description: (required for new tables only)

Table Volume: (required for new tables only, enter estimated volume for production)

Table Daily CRUD (Creates, Reads, Updates, Deletes/ required for new tables only):

C= _____ R= _____ U= _____ D= _____

(designate primary key columns for new tables)

Column Name	Data Type Nullity	Length	Action	Description (required for new columns only)

Notes:

- Data Type/Nullity - C=Character, I=Integer, S=Small Integer, N=Numeric / Y/N for Nulls Allowed/Disallowed
- Length - “n,s” n = whole number, s = precision, Length not required for Integer or Smallint Data Types
- Action - D=Drop, A=Add, C=Change

Please do not write below this line (DBA use only)

Impact Analysis:

Data Model Change Required? ____ Yes/No

Migration Record:

From	To	Approved / Date	Scheduled	Completed
Dev	Test	_____ _____		
Test	UAT	_____		
UAT	PROD	_____		

9.1.4 Quality Assurance Review

QA Review ID: ____

Program: _____

Author: _____

Review Date: _____

Reviewers: _____

Quality Assurance Analyst Approval:

Date: _____

Action Item	Completed

9.1.5 Move Request Form

Configuration Identifier: ____

Date: _____

Item: _____

Emergency? Yes / No

Release: _____

Emergency Auth: _____

Type:

Program _____
 Copybook _____
 CICS Map _____
 DCLGEN _____
 JCL _____
 JCL Control Cards _____
 Bind Cards _____
 DBRM _____
 Batch Load Module _____
 CICS Load Module _____

Notes:

From	To	Approved / Date	Completed
Development	System / Integration Test	_____ _____	
System / Integration Test	Acceptance Test	_____ _____	
Acceptance Test	Production	_____ _____	

9.2 Deliverables Acceptance Criteria

The following tables establish the preliminary acceptance criteria for each payment deliverable.

Deliverable	Deliverable Contents/Acceptance Criteria
WD TIP Phase I	
Updated Stakeholder Communication Plan	<ul style="list-style-type: none"> • Project stakeholders and classifications according to communication types • Communication processes that will be used to inform project stakeholders according to communication type • Process of communication development, review and approval prior to distribution • Conformity with the SAWS Communication Plan
Project Management Plan	<ul style="list-style-type: none"> • Project management standards and approach • Project scope • Scope control process • Roles and responsibilities • Project assumptions • Project workplan <ul style="list-style-type: none"> ⇒ Sequence and duration of tasks to be performed ⇒ Deliverables associated with the tasks ⇒ Resources required to perform each task ⇒ Dependency and constraint relationships between task ⇒ Scheduled start and end date for each task ⇒ Resource estimates for each task ⇒ Tracking of completed tasks • Project organization • Risk management plan including identification of existing risks and how they will be mitigated • Change control process • Issue resolution process • Internal quality assurance process • Automated project management tools
Monthly Status Reports	<ul style="list-style-type: none"> • Tasks completed during the reporting period • Tasks to be completed during the next reporting period • Issues requiring intervention • Earned versus burned hours calculation for the reporting period • Copies of project team member timesheets certified by the Project Manager
Configuration Management Plan	<ul style="list-style-type: none"> • Configuration management resource requirements • Establish authority and responsibility of configuration management to an identified resource

Deliverable	Deliverable Contents/Acceptance Criteria
	<ul style="list-style-type: none"> • Configuration management standards, procedures and guidelines, including processes to be used to establish baselines, control changes, record and track status and audit the product as well as the process through which versions of the TRAC Application releases will be prepared and documented • Project items subject to configuration management • Configuration management tools, including the repository for storing items and records
Updated Business Requirements Document	<ul style="list-style-type: none"> • Well defined and correctly stated business need statement directly tied to the applicable State and Federal Regulations • Clearly stated, verifiable, traceable requirements addressing the problem/business need • All requirements to adequately address the business problem • Assumptions • Issues • Consistency with the Project Management Plan • Ranking of requirements based on priority, criticality, feasibility, risk, source and type
Design and Coding Standards Guide	<ul style="list-style-type: none"> • Design standards <ul style="list-style-type: none"> ⇒ Rules for packaging code into sub-routines ⇒ Error handling and exception processing ⇒ Restart standards • Coding standards <ul style="list-style-type: none"> ⇒ Guidelines for using specific types of data (numbers, integers, floating-point numbers, characters, strings, etc.) ⇒ Rules for use of comments and audit trails • User interface/navigation standards • Screen standards • Screen help standards • Unit testing standards
System Architecture Model	<ul style="list-style-type: none"> • System architecture model • Technical architecture including interface design • Capacity plan
Implementation Strategy	<ul style="list-style-type: none"> • Implementation assumptions • Roll-out strategy • Communication strategy • Training strategy <ul style="list-style-type: none"> ⇒ Objectives ⇒ Approach ⇒ Readiness requirements • Conversion strategy

Deliverable	Deliverable Contents/Acceptance Criteria
	<ul style="list-style-type: none"> • Change leadership strategy • Help desk strategy
Detailed Design Specification Document	<ul style="list-style-type: none"> • Design overview <ul style="list-style-type: none"> ⇒ Traceability matrix of major components against requirements and functional specifications ⇒ Design assumptions • System interfaces • Input/output file layout, including data formats and data record lengths • Report layout • Data dictionary and data models • External Developers Guide comprised of a combination of design specification components • Edit rules • Error messages and exception processing • Restart procedures • Screen layout • TRAC Application architecture model • Reference codes • Backup/recovery plan • System security plan • Conformity with standards • Project construction tools (programming environments, source code, code libraries, code generators)
WDTIP Phase II Workplan	<ul style="list-style-type: none"> • The categorization, sequence and duration of tasks to be performed • The deliverables associated with the tasks • Dependency and constraint relationships between tasks • Resources required • Start and end date for each task • Identification of critical path elements • Identification of milestones and deliverables
WDTIP Phase II	
Monthly Status Reports	<ul style="list-style-type: none"> • Tasks completed during the reporting period • Tasks to be completed during the next reporting period • Issues requiring intervention • Earned versus burned hours calculation for the reporting period • Copies of project team member timesheets certified by the Project Manager
Completed Source Modules/Unit Test	<ul style="list-style-type: none"> • The functionality provided by each module • Completeness of each module

Deliverable	Deliverable Contents/Acceptance Criteria
	<ul style="list-style-type: none"> Relationship between the module and other source modules Full traceability to business requirements and design documents Conformity with Design and Coding Standards Guide deliverable Unit test checklist System test plan, scenarios and acceptance criteria
Implementation Plan	<ul style="list-style-type: none"> Roll-out plan Communication plan Training plan Conversion plan Help desk plan
Integration/System Test Sign-off	<ul style="list-style-type: none"> Summary of findings Description of tests conducted and system results for each test Signature and dates User acceptance test plan and scenarios
Training Curriculum	<ul style="list-style-type: none"> Training materials User documentation Conformity with the training plan
WDTIP Phase III Workplan	<ul style="list-style-type: none"> The categorization, sequence and duration of tasks to be performed The deliverables associated with the tasks Dependency and constraint relationships between tasks Resources required Start and end date for each task Identification of critical path elements Identification of milestones and deliverables
WDTIP Phase III	
Monthly Status Reports	<ul style="list-style-type: none"> Tasks completed during the reporting period Tasks to be completed during the next reporting period Issues requiring intervention Earned versus burned hours calculation for the reporting period Copies of project team member timesheets certified by the Project Manager
User Acceptance Test Sign-off	<ul style="list-style-type: none"> Summary of findings Description of tests conducted and system results for each test Signatures and dates
Conversion Process and Procedures	<ul style="list-style-type: none"> Conformity with conversion plan Encountered problems for each county Maintenance and review of data integrity

Deliverable	Deliverable Contents/Acceptance Criteria
User Training	<ul style="list-style-type: none">• Conformity with the training plan• User understanding of the enhanced system
System In Production	<ul style="list-style-type: none">• System documentation• Operations guide• Transition plan (M&O)• Certification (Deloitte Consulting)